NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA

THESIS

DISTRIBUTED CONTROL FOR PARALLELED PWM INVERTERS IN THE DC ZONAL DISTRIBUTION SYSTEM

by

Mark L. Marinac

September 1999

Thesis Advisor: John G. Ciezki
Co-Advisor: Robert W. Ashton

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 1999	Master's Thesis

4. TITLE AND SUBTITLE
Distributed control for paralleled PWM inverters in the dc zonal distribution system

5. FUNDING NUMBERS

6. AUTHOR(S)
Mark L. Marinac

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT ***(maximum 200 words)***

The modular DC Zonal Electrical Distribution System (DC ZEDS) offers many advantages over traditional radial shipboard electrical distribution. The advantages of DC ZEDS will be exploited in the next class of surface combatants. Part of the development research for DC ZEDS includes the design of autonomous DC-to-AC inverter modules having robust load sharing capability. This thesis will focus on these DC-to-AC inverters.

Paralleled inverters in the DC ZEDS must be flexible and modular. The optimum flexibility is achieved when an electrical system can automatically reconfigure after the number of operating inverters changes. A distributed control system for paralleled inverters is investigated to achieve this flexibility and increase the system's battle damage survivability. Modularity of the system design allows for simplified testing and a reduction in cost through bulk purchasing. Unique distributive control methods are developed in this thesis that will allow isolated inverters in separate watertight compartments to coordinate their actions without interconnections between the units.

14. SUBJECT TERMS	15. NUMBER OF PAGES
DC-to-AC PWM Inverter, Synchronous Reference Frame Analysis, Power Line Communication, Droop-Based Load Sharing, Paralleling, ACSL	116
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18, 298-10

Approved for public release; distribution is unlimited.

DISTRIBUTED CONTROL FOR PARALLELED PWM INVERTERS IN THE DC ZONAL DISTRIBUTION SYSTEM

Mark L. Marinac
Lieutenant, United States Navy
B.S., University of Texas at Austin, 1992

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 1999

Author: _____
Mark L. Marinac

Approved by: _____
John G. Ciezki, Thesis Advisor

_____
Robert W. Ashton, Co-Advisor

_____
Jeffrey B. Knorr, Chairman
Department of Electrical and Computer Engineering

ABSTRACT

The modular DC Zonal Electrical Distribution System (DC ZEDS) offers many advantages over traditional radial shipboard electrical distribution. The advantages of DC ZEDS will be exploited in the next class of surface combatants. Part of the development research for DC ZEDS includes the design of autonomous DC-to-AC inverter modules having robust load sharing capability. This thesis will focus on these DC-to-AC inverters.

Paralleled inverters in the DC ZEDS must be flexible and modular. The optimum flexibility is achieved when an electrical system can automatically reconfigure after the number of operating inverters changes. A distributed control system for paralleled inverters is investigated to achieve this flexibility and increase the system's battle damage survivability. Modularity of the system design allows for simplified testing and a reduction in cost through bulk purchasing. Unique distributive control methods are developed in this thesis that will allow isolated inverters in separate watertight compartments to coordinate their actions without interconnections between the units.

TABLE OF CONTENTS

I. BACKGROUND FOR THE PROJECT

The Navy is investigating ways to apply innovative technology and commercially available products in order to create sophisticated yet cost effective ships. Designers are facing the challenge of meeting the operational requirements of the Navy with increasingly tight budgets. In general the Navy desires new designs to be flexible, reduce manning requirements, maximize effectiveness, and enhance survivability all at a reduced cost. The power generation and distribution system offers one area for improvement in shipboard design. If it is possible to reduce the amount of cabling, reduce the weight of cabling, and minimize the size of the generation and conversion equipment, then that would be advantageous. But, by reducing the amount of cabling and switchgear components, survivability or reliability must not be sacrificed. To fully understand the need for a new inverter control design, the Navy's current electrical system paradigm is examined, and then the proposed DC Zonal Electrical Distribution System (DC ZEDS) is introduced.

A. TYPICAL SHIPBOARD ELECTRICAL DISTRIBUTION SYSTEM

The typical shipboard electrical distribution system derives its power from a small number of large capacity alternating current generators. AC generators are preferred to DC generators because they are generally smaller and do not require as much maintenance as their DC counterparts. The generators are located in the center and aft positions of the ship, typically in the lower decks near the main engines. The power produced by the generators is then fed to switchboards that distribute the power to loads and other feeder switchboards. This switchboard hierarchical design is known as radial distribution. Thus, for radially distributed electrical systems, you will find a tree of switchboards and loads from each generator. To complicate matters, some switchboards receive power from more than one source. Additionally, two separate trees may be joined through isolations to vital loads. Vital loads receive their power from switchboards with

different generator sources normally using an auctioneering scheme. If one generator or switchboard tree is out of service, these vital loads remain powered by the auxiliary path generator.

The main advantages of the radial design is that it is an operationally tested and working design, and it provides redundancy and reliability. There are many disadvantages of the radial design. The radial design complicates power isolation due to its serial-tree distribution system. For casualty control, isolating an 'upstream' switchboard or breaker may unnecessarily isolate many additional loads not in the effected space. The radial design requires redundant cabling, with two or more branches traversing the ship. This branching leads to more weight, more bulk, and more watertight compartment penetrations. Radial type distribution systems are not standardized. Switchboards and the breakers that isolate them are located at different places on different ships. One must rely on ship plans and manuals in order to trace systems for tag out or casualty isolation. Many times these sources of information are not updated as quickly as modifications are performed to the ship making isolation tricky. Also, a nonstandard design means that maintenance and operations personnel require additional training with every duty station change.

B. ZONAL POWER DISTRIBUTION

Zonal power distribution is an alternative to the current radial design. This design addresses many of the problems associated with the radial distribution system. In a zonal distribution scheme, power is distributed through the ship by at least two parallel longitudinal busses. These busses could be separated into port and starboard busses to improve their survivability. The busses could further be optimized by having some located above the waterline to protect against flooding and some located below the waterline to provide maximum shielding from bombardment and aerial attack.

Instead of a tree-type structure, the electrical system would be divided into zones normally defined by watertight compartment boundaries. This scheme has many

advantages. The zonal structure allows for easy isolation of power to a single watertight compartment, and combined with the longitudinal busses, ensures that every watertight compartment will normally only have two electrical penetrations. This improves the space's watertight integrity. The busses will also eliminate significant cabling thereby allowing a substantial weight reduction and space savings.

Finally, the zonal structure allows for standardized design and testing. Technicians from any class of ship will be able to learn the system more quickly and should be able to transition ships and already know how to isolate a zone in a casualty situation. During construction, the electrical system could be tested and verified in modules.

C. DC ZONAL DISTRIBUTION

The zonal distribution scheme has many advantages but still needs development. A question remains as to whether the distributed power should be AC or DC. It is customary to use AC power distribution since that is what utilities use to transmit power over long distances where transformers facilitate convenient voltage-level shifting. Since the ship is much more compact, DC distribution becomes an option. Thus, the possibilities of a DC Zonal Electrical Distribution System (DC ZEDS) become feasible.

The planned layout for the DC ZEDS requires at least three main multi-phase Ship Service Generators (SSGs) supplying the system. Each generator's output will be immediately converted into DC using a Phase Controlled Rectifier (PCR). A high DC voltage from the PCR is desired to minimize the current in the main bus thereby allowing the bus cabling to be smaller and lighter. Presently, the DC voltage output of the rectifier is limited to 1600V and 800A due to the limits of the Insulated-Gate-Bipolar Transistors (IGBTs) and MOS-Controlled Thyristors (MCTs) in the conversion equipment [1]. Also, higher voltages introduce grounding concerns as well as the need for isolation in the inverter modules.

The resultant DC power is fed to the respective longitudinal bus for distribution throughout the ship. Ships Service Converter Modules (SSCMs), shown in Figure 1, step down the main bus voltage in each watertight compartment or zone. The lower voltage tightly regulated DC power can then be used to directly supply dc loads, supply additional SSCMs which provide even lower DC voltages for use by combat systems, or supply DC power to Ship Service Inverter Modules (SSIMs). The SSIMs are solid-state DC-to-AC soft-switched Pulse Width Modulated (PWM) inverters. The SSIMs produce all the ac power required by the zone. Typically they will produce three-phase 450V AC power, but they will also supply the ship's 400 Hz loads and variable frequency electric drives.

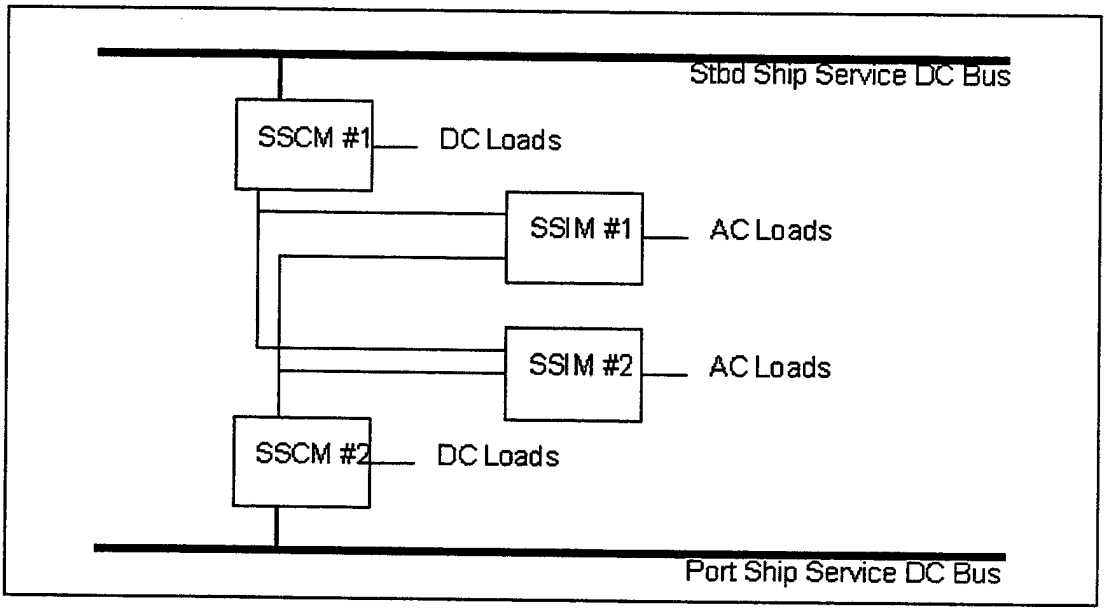


Figure 1. A Simplified Block Diagram of a DC ZEDS Zone.

It is important to note that AC power is being produced from a DC source. Thus, the AC power output of the SSIMs is not dependent on the frequency of the SSGs. This de-coupling of the AC loads means that the gas turbines driving the SSGs can be operated at their most efficient speeds, saving fuel and reducing emissions.

With DC power, the job of paralleling the port and starboard busses reduces to simply matching their voltages. With AC power one must match the voltage, frequency,

sequence, and phase in order to successfully parallel the sources. The SSCMs would accomplish isolation for the port and starboard busses during paralleling. These conversion modules would buffer the main buses from faults, while stepping down the high bus voltages to usable levels for the zone. With this isolation and ease of paralleling, many loads could be powered from both busses and the load on the electrical system could be equally divided between the busses.

With DC power, variable-speed high-torque motors could be used. The ability to vary the speed of the motor while retaining its maximum torque greatly increases the efficiency of the motor for large loads. The operation of a large number of hydraulic pumps and water pumps on board, which are constant speed or use separate windings for speed control, could be optimized.

DC ZEDS uses a number of small solid-state converters to control variable-speed drives instead of the bulkier Motor-Generator (MG) sets currently in use. The inherent buffering ability of these converters means that circuit breakers will be supported by intelligent monitoring and protection.

D. PARALLELED INVERTERS

Inverter-based power conversion methods address many of the problems inherent to traditional DC-to-AC conversion methods used in the Navy. Currently, large-scale DC-to-AC conversion is typically carried out using a DC machine mechanically coupled to an AC machine. The key advantage to this technology is that it is tried and tested to work on ships and submarines. But, there are many disadvantages to this type of power conversion. The first disadvantage is the response time to large transients. The physical construction of the machine puts limits on its response time due to issues of torque and momentum changes. In addition, these machines are significantly larger than a comparable inverter system. Finally, with a large DC machine, extensive maintenance is required for the upkeep and cleaning of the brushes. Thus, by using the solid-state inverter technology we gain a faster, smaller system that requires less maintenance.

Additional advantages may be realized from an inverter-based power conversion system when the inverters are operated in parallel. DC ZEDS is desirable due to its compact DC-to-AC conversion system for ease of operation and maintenance, but at the same time it must be flexible with the ability to automatically reconfigure after device failure or battle damage. Having many smaller inverters in parallel is more reliable than a single large inverter [2]. Designing these small inverters to operate without a centralized control scheme also increases system reliability.

A decentralized, parallel design reduces the system's obsolescence. The ultimate goal would be to design a "plug and play" inverter. This inverter would have a self-contained controller and output filter. Installation would simply require access to the AC bus of interest and access to a source of DC. Therefore, to compensate for ship upgrades requiring more AC power from the bus, an additional inverter is simply attached to the appropriate bus. By making all the inverters identical, individual unit cost could be reduced and repair parts would potentially be more accessible. Thus, by using paralleled inverters a more robust and cost effective power system is made available on naval ships.

E. THESIS GOALS

In this thesis a computer model for paralleled three-phase SSIMs is developed, a control system is designed for the SSIMs, and an information sharing system for the SSIMs is developed. The design must be modular and robust. Simulations and practical experiments are used to substantiate the design.

F. CHAPTER OVERVIEW

Chapter II documents the development of the design. Initially, a review of inverter theory is presented followed by a description of the specifications and constraints on the design. Chapter III contains a description of the assumptions and techniques used to model the operation of an individual inverter on a computer. The nonlinear aspects of the solid-state switching operation model are carefully examined.

In Chapter IV the complexity of inverter load sharing is addressed. Inverter topology considerations are also covered. In Chapter V inverters are paralleled using an autonomous voltage-droop technique. Finally, recommendations for future study and conclusions are set forth in Chapter VI.

THIS PAGE INTENTIONALLY LEFT BLANK

II. DEVELOPMENT OF THE DESIGN

A. INVERTER THEORY

Inverters are solid-state devices used to produce AC voltage and current from a DC source. To understand how the inverter operates one must understand the purpose of switches in the design and the reason for output filtering. To understand the use of the inverter in a particular system, one must also know the different inverter configurations. This section addresses these topics.

1. Using Switching to Produce AC

To transform the DC voltage and current into an AC voltage and current, solid-state switches are used to discretize the DC. These switches are normally FETs, BJTs, and IGBTs, though new MCT devices are also coming onto the market. Current technology limits IGBTs to 800 amps and a switching frequency of about 50 kHz [1]. A basic three-phase inverter topology is shown in Figure 2.

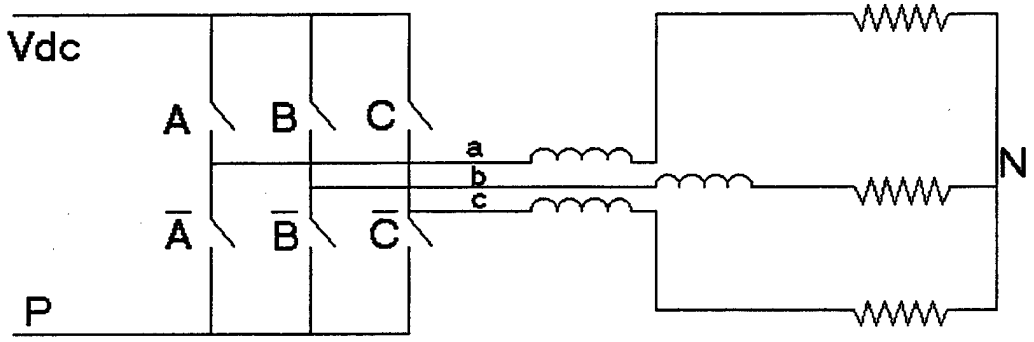


Figure 2. The Basic Three-Phase Inverter Topology.

The inverter model is divided into three physical groups: the switching group, the filter group, and the load group. The switching group consists of the DC supply, the switches, and the switch diodes. The DC supply can be realized with batteries or the filtered output of an AC-to-DC rectifier. For the topology illustrated in Figure 2, there are six switches required for a three-phase system. To synthesize a stair-stepped approximation of a sinusoidal output, the inverter switches must be capable of being configured to produce alternately a positive, negative, and a zero load line-to-line voltage. In order to implement the voltage switching and accommodate bi-directional load current flow, two switches per phase are required. To illustrate these points assume that switch A is shut and switch $\overline{B}$ is shut with all other switches open. The path of current flow is from $+V_{dc}$ through Switch A, through the inductor and resistor in phase 'a', to the common point N, through the resistor and inductor in phase 'b', through Switch $\overline{B}$ to the common point P. Thus the switch arrangement determines the current-flow direction and the voltage convention at the load. If all the switches are now opened and then Switch $\overline{A}$ is shut and Switch B is shut, the current flow through and potential difference across the load reverse. Therefore, from a single DC source a positive and negative current flow through the load has been created along with a sign change in the potential across the load. For non-unity power factor loads, diodes must be included across the switches to provide current paths for energy storing loads.

Note that the two switches in the same phase cannot be shut at the same time or the DC power supply is short circuited. To prevent this shorting, the switches on a single leg (i.e. A and $\overline{A}$) are always switched in a complimentary fashion. Thus, instead of six individual switches, there are three sets of switches with two mutually exclusive positions per set. This fact leads to eight possible switch combinations per inverter as illustrated in Table II-1, where the switch position refers to the top switch.

State	Switch A	Switch B	Switch C
1	Shut	Open	Open
2	Shut	Shut	Open
3	Open	Shut	Open
4	Open	Shut	Shut
5	Open	Open	Shut
6	Shut	Open	Shut
7	Shut	Shut	Shut
8	Open	Open	Open

Table II-1. Switching States for a Three-Phase Inverter.

In [7] Mohan demonstrates that for balanced loads, the phase voltages can be determined solely by the switch positions as shown in Equations (1) through (3). V_{ap}, V_{bp}, and V_{cp} are equal to the DC supply voltage if that phase's switch is shut and zero if the switch is open.

$$V_{Phase\_a} = \frac{2}{3}V_{ap} - \frac{1}{3}\left(V_{bp} + V_{cp}\right) \quad (1)$$

$$V_{Phase\_b} = \frac{2}{3}V_{bp} - \frac{1}{3}\left(V_{ap} + V_{cp}\right) \quad (2)$$

$$V_{Phase\_c} = \frac{2}{3}V_{cp} - \frac{1}{3}\left(V_{ap} + V_{bp}\right) \quad (3)$$

From the equations, the following discrete phase voltage levels are possible: -2/3 V_{dc}, -1/3 V_{dc}, 0 V_{dc}, 1/3 V_{dc}, and 2/3 V_{dc}. This discrete, stepped waveform is rich in harmonics. Appropriate switch modulation and a sufficiently high switching rate may shift the harmonics to frequencies higher than that of the desired fundamental voltage. However, semiconductor device limitations prohibit arbitrarily high switching rates and some output filtering is always required.

2. Output Filtering

The output filter of the inverter is the second physical group of an inverter system. This filter is required to pass the desired fundamental frequency produced by the discrete inverter output. By performing this task, the filter performs two desired functions. First, it smoothes the stepped DC voltage levels produced by the inverter into an AC signal. Second, it passes the desired frequency power while significantly attenuating harmonics, undesired frequencies, and noise from the system.

Complicated filter design is not required in many cases. If the inverter is designed with a switching frequency far above the fundamental frequency, most of the undesirable frequencies will lie at or above the switching frequency [7]. This fact allows the use of a low pass filter to produce the desired results for a system with switching frequencies in the kilohertz range. Including a capacitor after each output inductor in Figure 2 would create a low pass filter. The designer of this filter must be careful to anticipate the types of loads connected to the inverter. Energy storage elements in the load will likely alter the filter behavior [8]. Adequate filter design allows the fundamental frequency to remain in a region with a gain of 0 dB and the switching harmonics to be reduced by at least 40 dB for all anticipated load types.

3. The Load

The final physical group of the inverter system is the load. Unlike the switching hardware and the filter, the load is the dynamic part of the system. In the Figure 2 the load consists of a simple resistor. On naval vessels the load is more likely to contain large inductive elements and possibly constant power sinks. These energy storage elements can make the control of the inverter much more difficult since both real and reactive power must be managed and for constant power loads, negative input impedance effects introduce stability issues.

B. STAND-ALONE INVERTER CONTROL REQUIREMENTS

The inverters under consideration in this thesis are three-phase, voltage source, PWM inverters. The controller for this inverter must be developed to satisfy a set of minimum requirements that are identified in this section. In future sections, alternative solutions will be contrasted using these criteria.

1. Bus Parameters

The controller must be able to ensure that the inverter output voltage and frequency satisfy the manufacturer specification of all the anticipated loads. The parameters used for this thesis are:

1. The bus voltage: 450 V ± 10%
2. The bus frequency: 60 Hz ± 0.5%
3. Input Voltage: 950 V ± 20 V [11]

The bus voltage and bus frequency values are based on typical naval bus parameters. The input voltage values were obtained from an NPS Master's thesis by Moore [11] involving SSCMs.

2. Response Time

In addition to maintaining the bus parameters in steady-state conditions, transient response to a changing load is also important. Typically it is desired that the controller be as fast as possible with a minimum amount of over or undershoot after the transient load change. The response time should be at least an order of magnitude less than the switching frequency and an order of magnitude greater than the fundamental frequency. Thus for a 60 Hz bus frequency and a switching frequency of 10 kHz, a settling time of approximately one millisecond would be satisfactory.

THIS PAGE INTENTIONALLY LEFT BLANK

III. MODELING AN INDIVIDUAL PWM INVERTER

In this chapter, background on the development of a computer simulation of a stand-alone three-phase inverter is discussed. Then, the design for this inverter model and its controller is explained. Finally, results from testing this stand-alone three-phase inverter are presented.

A. THE INVERTER DESIGN GROUNDWORK

A software model was developed in lieu of a hardware model for this thesis. In this section the rationale for this decision is explained.

1. Simulation Tools

Simulation is an excellent means to perform initial testing of a design prior to investing time into construction and capital into parts. For this thesis detailed simulations were developed to demonstrate proof of concept. The two main simulation tools used to build the inverter model were the Advanced Continuous Simulation Language (ACSL) and Matlab. Matlab was particularly useful in testing individual pieces of the design prior to incorporating it into the main model module. The inverter output filter was exclusively designed using Matlab. Later in the design process, Matlab's system design tools were used to test the open and closed-loop poles of the system and to determine controller gains to ensure adequate system response. Although ACSL can perform these same tasks, Matlab was more convenient and faster to code. Matlab was easier to use since it is PC based and it operates as an interpreted language. The version of ACSL that was available for use in this thesis is a UNIX-based program that must be compiled and executed.

ACSL was used to verify the overall design. This language has many very powerful features and few weaknesses. ACSL was designed to model continuous systems described by time-dependent, non-linear differential equations. One of its

strengths is that the program is self-sorting. Equations can be entered in any order and ACSL will execute them in the proper sequence such that the output from one equation is available to the input of the next equation.

ACSL performs superlative to programs such as Matlab for handling systems of differential equations. The equations do not have to be entered as a matrix, just entered the way they appear on paper. Using a simple INTEG statement, ACSL will find the state variables for each step in the simulation. For more control, the user can set the maximum and minimum step sizes allowed for the simulation.

ACSL also contains powerful discrete functions. These functions allowed the inverter switches to be modeled more accurately. To increase the accuracy of a discrete event ACSL iterates around an event to ensure the state variables are correct. Without this discrete ability, the switches would have to be averaged out of the model. This averaging process would reduce the model's flexibility and level of detail.

ACSL's discrete iteration improves accuracy, but it comes at a price. Algebraic loops are not permitted. An equation of the form Y= sin(Y*x) would introduce an error. Implicitly, an equation of the form Y = f(Y) can be solved in ACSL using the Implicit Command (IMPL) as long as f(Y) can be written as a closed-form equation. Also, the use of IMPL reduces code execution speed dramatically. In the case of the inverter there are many discrete events which make the statement f(Y) not realizable. Thus, a simple feedback loop for a proportional controller becomes difficult to build. This is ACSL's most noticeable weakness, but it can be worked around. Using discrete scheduling or IF-THEN blocks the programmer can incorporate implicit routines into the code without causing an error. The disadvantage to this workaround is that the execution speed will slow down and the code can yield less accurate results.

2. Hardware Limitations

Limited hardware availability was another reason simulation was performed. Although enough hardware was available to build stand-alone and paralleled inverters in

the lab, convenient DSP control was not available for more than a single unit. This meant that the inverters could be implemented, but only with analog control. The overall focus of this thesis is the design of a controller for paralleled inverters, therefore the constraint of analog controls was deemed unsatisfactory. Currently ordered hardware (dSPACE) will allow further experimentation using DSP control.

A single three-phase inverter was built using the Lab-Volt EMS 8837 Power MOSFET module. The controller used was the Lab-Volt EMS 9029 Chopper/Inverter Control Unit. Using these components the actual output of a PWM inverter could be observed and recorded from an oscilloscope. The outputs from the computer simulation could then be compared to these 'real world' results to verify the model's validity.

B. THE SYSTEM DESIGN

As stated in Chapter II, the inverter being considered is a three-phase, voltage source, PWM inverter. Four aspects of the inverter design must next be discussed: choosing the inverter type, designing the system filter, representing the load, and modeling the system.

1. Choosing the Inverter Type

Sine-triangle modulation is the most widely adopted and understood method to generate PWM signals and is simple to produce in analog or digital hardware. For these reasons sine-triangle modulation was selected over space vector modulation for this project.

Superimposing a control sinusoidal waveform over a triangle waveform develops the switching scheme for sine-triangle modulation as shown in Figure 3. Whenever the sinewave's amplitude is greater than the triangle's, the top switch in an inverter leg is shut and the bottom switch is left open. When the sinewave's amplitude is smaller than the triangle's, the top switch is open and the bottom switch is shut. The control

sinewaves for each of the phases of the three-phase inverter are operated 120 degrees out of phase from the others to produce the desired balanced waveforms.

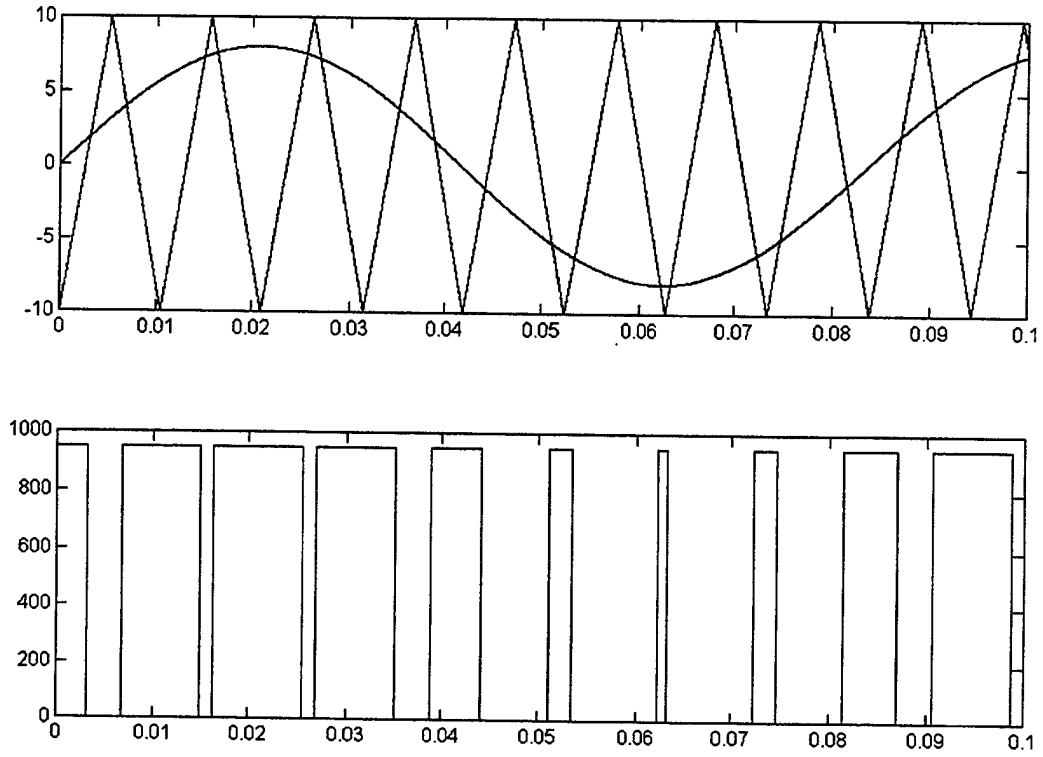


Figure 3. The Sine-Triangle Waveform and its Corresponding PWM Output.

The triangle waveform's frequency sets the switching frequency of the inverter. The switching frequency is the rate at which the solid-state switches are gated; it is not the output fundamental frequency. As discussed in Chapter II, the switching frequency should be an integer multiple of double the fundamental frequency in order to minimize output harmonics. For this model a switching speed of 10,800 Hz was used (60 Hz * 2 *90=10,800 Hz). Higher switching frequencies (up to 50 kHz) are available, but these frequencies unnecessarily increase the simulation time with little new information gained on the success of the implemented algorithms. A switching frequency of 10.8 kHz is high enough to give adequate separation from the fundamental frequency, allowing the use of a simple low-pass filter on the inverter outputs.

The superimposed control sine wave sets the voltage and frequency of the inverter's AC output. The fundamental phase-voltage output amplitude V_o from the sine-triangle inverter operating in the linear modulation range ($V_c < 0.5V_{dc}$) is shown in Equation (4). V_o is dependent on the triangle peak voltage, V_{tri}, the DC source voltage, V_{dc}, and the instantaneous sinusoid control voltage, V_c.

$$Vo = \frac{Vc}{Vtri} \cdot \frac{Vdc}{2} \tag{4}$$

The control sinewave's frequency sets the fundamental frequency directly. With the frequency set, a choice must be made to regulate the inverter output voltage or current. On naval ships the AC loads are typically constant voltage type loads; therefore, the controlled outputs should be the bus voltages.

2. Designing the System Filter

The system filter is fundamentally important to the operation of the inverter. As discussed in Chapter II, the raw output of the inverter is a stepped or pulsed waveform which is rich in harmonics, particularly at the switching frequency and integer multiples of the switching frequency. A filter is inserted at the output to reduce these harmonics. The filter is represented by the RLC components depicted in Figure 4.

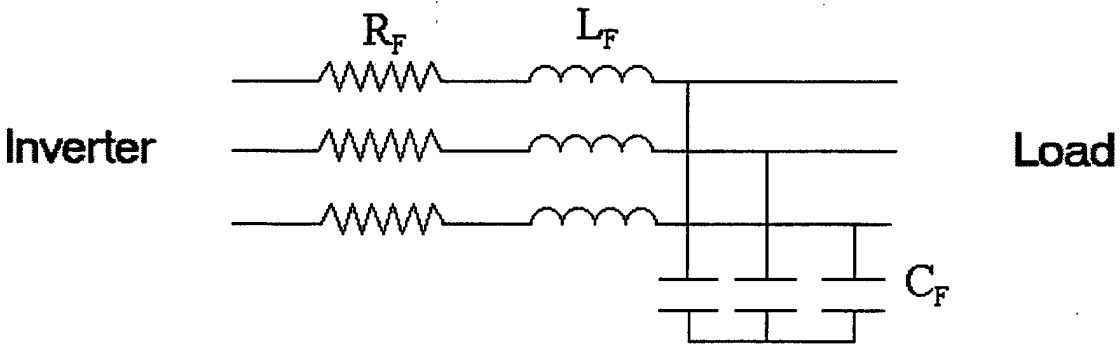


Figure 4. The Basic Three-Phase Filter Topology.

The output resistance R_F is included in the filter to model the non-ideal resistances in the switches. The effect of the line impedance and the load on this filter are detailed later. The resulting transfer function for the low-pass RLC filter is shown in Equation (5).

$$\frac{Vout}{Vin} = \frac{1}{L_F C_F s^2 + R_F C_F s + 1} \tag{5}$$

The filter must pass the fundamental 60 Hz power unaltered while attenuating frequencies at and above the 10 kHz switching frequency. Additionally, in order to accommodate possible signal injection methods in the future, the cutoff frequency for the filter was set at 200 Hz. Using this cutoff frequency in Matlab's Butterworth Filter program and a filter resistance $R_F = 0.01\ \Omega$, the values of C_F and L_F were found to be 3.75E-4 F and 1.878E-4 H respectively.

Butter(2,200*2*pi,'s') (6)

The filter provides a maximally flat, near 0 dB response (0.1 dB) at the fundamental frequency, while attenuating the switching harmonics by over 50 dB as seen from the Bode plot of Figure 5.

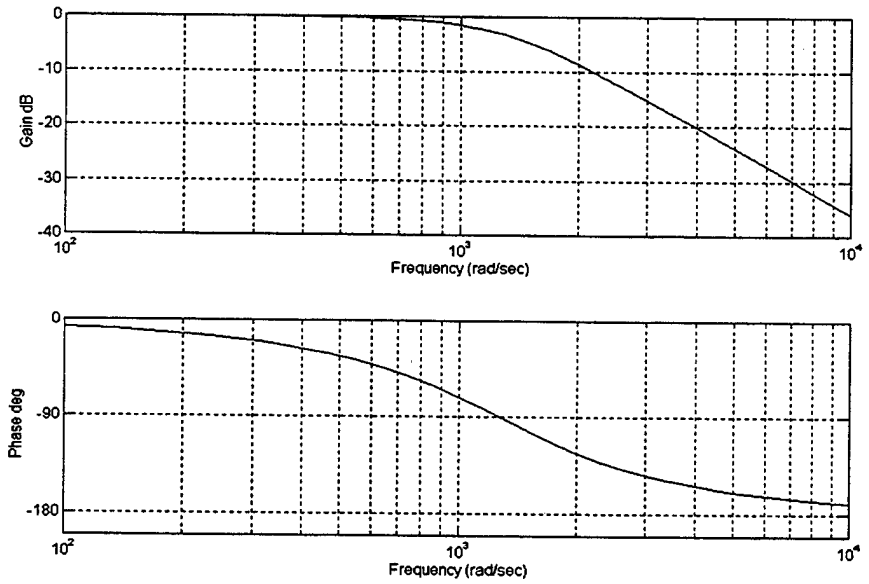


Figure 5. Matlab Bode Plot of the System Filter.

3. Representing the Load

The load model is shown in Figure 6.

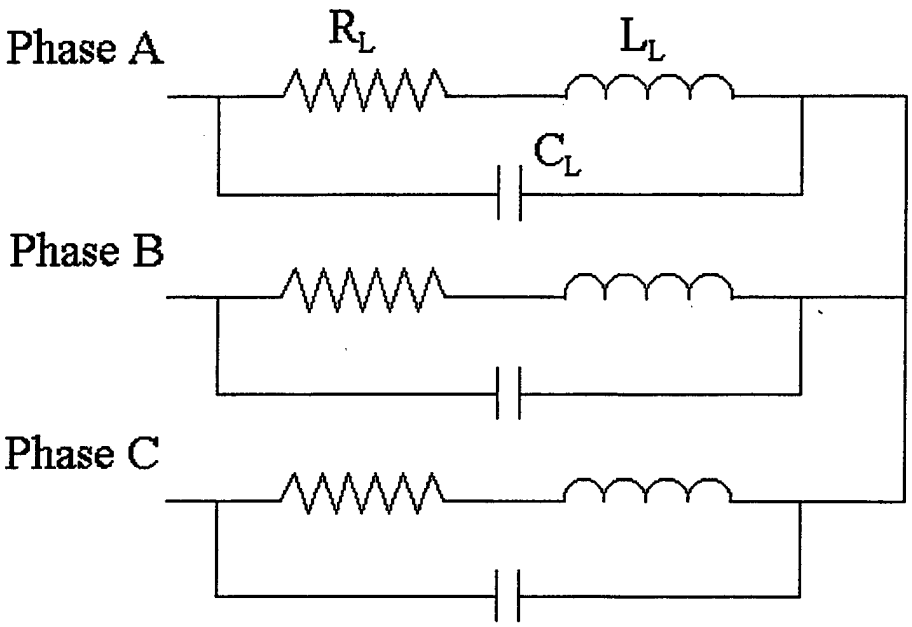


Figure 6. The Load Representation.

Typical values for the load of R_L=20 Ω, L_L=0.021 H, and C_L=10 μF were chosen as a starting point. This load is more complicated than loads referred to in the literature [2]-[6]. Most authors assume a simple resistive load or a series RL load. Including the capacitance of a load makes this model more complete and flexible. But, the additional energy storage device increases the complexity of the analysis and the run time for computer simulations. Also, unlike a purely resistive load, the RLC load can adversely influence the operation the inverter's filter. By using a worst-case load, the robustness of the filter design and the controller can be verified. These effects are studied in the analysis portion of this chapter.

4. Modeling The System

A complete system model for the stand-alone inverter can now be set forth. This system is divided into a discrete section and a continuous section. The discrete portion models the switching action, and the continuous portion models the filter, the lines, and the load. The development of equations for the continuous portion of the system is explained first, with that section of the system illustrated in Figure 7. This model can be simplified by observing that for a balanced network where Phase A, B, and C all have equal loading and the inverter is symmetrically fired, the neutral points N_F and N_L are at the same potential. Thus, for purposes of analysis these two points can be electrically connected [7].

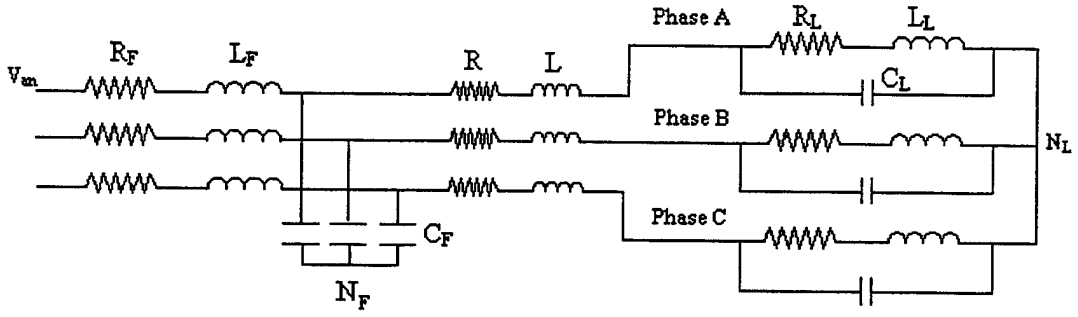


Figure 7. The Single Inverter System.

The potential in a phase is typically measured from the switch to the neutral point. Therefore we can represent an individual phase as a voltage from the switch to N_L. Chapter II explained that for balanced loads the phase voltages are determined based solely on the switch positions. V_{an}, V_{bn}, and V_{cn} represent the total voltage potential across each phase. The output that is to be regulated is the potential difference across the load capacitor, V_L.

The resulting equations for the phases differ only by the phase subscript, so only the equations for a single phase are developed here. The equations for Phase A are:

$$V_{an} = i_{Fa}R_F + L_F\frac{d}{dt}i_{Fa} + V_{Fa} \tag{7}$$

$$C_F \frac{d}{dt} V_{Fa} = i_{Fa} - i_{2a} \tag{8}$$

$$L_2 \frac{d}{dt} i_{2a} = V_{Fa} - V_{La} - i_{2a} R \tag{9}$$

$$C_L \frac{d}{dt} V_{La} = i_{2a} - i_{La} \tag{10}$$

$$V_{La} = i_{La} R_L + L_L \frac{d}{dt} i_{La} \tag{11}$$

The variable i_{Fa} is the a-phase filter current, i_{La} is the a-phase RL-interconnection impedance current, V_{La} is the a-phase load voltage, and i_{La} is the a-phase load current.

There are five energy storage elements in Phase A, so there will be five state variables per phase for a total of 15 states. Solving the equations for the derivatives of the state variables yields.

$$\frac{d}{dt} i_{Fa} = \frac{1}{L_F}\left(V_{an} - V_{Fa} - i_{Fa} R_F\right) \tag{12}$$

$$\frac{d}{dt} V_{Fa} = \frac{1}{C_F}\left(i_{Fa} - i_{2a}\right) \tag{13}$$

$$\frac{d}{dt} i_{2a} = \frac{1}{L_2}\left(V_{Fa} - V_{La} - i_{2a} R\right) \tag{14}$$

$$\frac{d}{dt} V_{La} = \frac{1}{C_L}\left(i_{2a} - i_{La}\right) \tag{15}$$

$$\frac{d}{dt} i_{La} = \frac{1}{L_L}\left(V_{La} - i_{La} R_L\right) \tag{16}$$

Matlab reveals that all of the eigenvalues for the open-loop system lie in the left-hand plane as shown in the table below.

-1.50E5 ± 2.79E5
-1.50E5 ± 2.78E5
-8.87E2 ± 1.27E3
-8.87E3 ± 5.16E2
-1.20E3 ± 3.77E2

Table III-1 Eigenvalues of the Open-Loop System for an Individual Inverter.

The eigenvalues show that the open-loop inverter is asymptotically stable. But, this data represents only one of many possible load configurations for the inverter. The problem of system configuration dynamics is further investigated when the system controller design is documented.

The discrete portion of the model relies on the controller output, but its general operation can be explained without knowledge of the controller. The sinusoidal output of the controller is compared to an internally generated triangle wave for each of the phases. An ACSL 'Schedule' command is used to determine when the sine wave and triangle wave cross. The Schedule command then calls a discrete section of the code that determines which switches should be shut or open. The Schedule command ensures that the timing for the switching and the times for the state variable integration are as accurate as possible. IF-THEN statements could have been used, but simulation accuracy is deteriorated.

C. DESIGNING THE INVERTER CONTROLLER

Controllers for solid-state inverters must be faster and more accurate than for typical AC generators. Inverter units have much lower output impedance than generators. Further, unlike a generator that has mechanical inertia, the inverter units can change their voltage very rapidly. The response of the inverter system is predominantly determined by the control systems [4].

1. Power Generating Equipment Control

Although inverters are power conversion devices, as far as the load is concerned inverters are simply AC sources, similar to an AC generator. Design of controllers for power generation/conversion equipment is not as straightforward as other control designs. Design of a controller typically involves modeling the system and developing the system's transfer functions. The controller is then tested using various inputs. Usually a PID type controller is tried first. Depending on the controller and the system, the designer may be able to place the closed-loop system poles in well-known locations (i.e. Bessel Roots). Iteration and refinement of these controller gains can usually yield good results.

Power generation equipment control is complicated since the system transfer function changes as the load changes. Mostly the load resistance and inductance are the parameters varying in the model. Therefore, tweaking a controller's gains for a specific load is not particularly useful since the poles of the system can and will change over time. Instead, the controller must be designed to give an adequate response for a range of loads or systems.

In addition to the control problem of a time-varying system, there is the added problem of the AC inverter feedback. Testing the system model with programs such as Matlab does not yield much information. The system was designed to operate with a sinusoidal input, so system tools such as STEP are not immediately useful in determining the system's stability. In addition, with the feedback and modulating signals being AC, phase concerns and beat frequency effects arise in the controller. These feedback signals could be filtered and rectified prior to being used in the control, but this technique adds additional dynamics to the system and slows the controller's response. An alternative approach is to use a more advanced control technique.

2. Synchronous Reference Frame Model

A controller for an inverter had to be designed which is testable using conventional control techniques and which did not require the combination of sinusoidal feedback quantities. Much of the literature on the topic of inverter control also recognizes this problem [2], [7]. A popular solution is to change the reference frame or coordinate system in which the system is modeled. One such representation is the transformation to the q-d-0 reference frame. In this reference frame, q, d and 0 are dummy variables derived from applying a transformation to the a, b, and c phase quantities of interest. For balanced loads in the synchronous reference frame, the zero-sequence quantities (0 terms) are identically zero and can be ignored [10]. Thus, the synchronous reference frame can be further simplified to just the q and d variables for this model. Although the remaining variables, d and q, do not represent any intuitive quantities, the q-d reference frame transforms sinusoidal variables into constants (at the synchronous reference frame frequency) when the system is in steady state. With constant forcing functions, the tools available for analyzing and optimizing conventional control systems may be used to test the response and quality of the designed controller. This is the power of the q-d transformation. A more detailed development of stationary and synchronous reference frame inverter control can be found in [10]. The development of the model for the inverter topology used in this thesis will mimic that found in [10].

The first task to perform before designing the controller was to convert the a-b-c system model into the synchronous reference frame. The transformation given in Equation (17) is applied to the state variable model derived for the a-b-c system.

$$K_s^e = \frac{2}{3}\begin{bmatrix} \cos\theta_e & \cos\left(\theta_e - \frac{2}{3}\pi\right) & \cos\left(\theta_e + \frac{2}{3}\pi\right) \\ \sin\theta_e & \sin\left(\theta_e - \frac{2}{3}\pi\right) & \sin\left(\theta_e + \frac{2}{3}\pi\right) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}\begin{bmatrix} q \\ d \\ o \end{bmatrix} \tag{17}$$

The zero-sequence terms are ignored as described above but are shown in the transformation for completeness. The angle θ_e is the phase angle of the transformation and is given by Equation (18). The variable *we* is the system's radian frequency and *t* is the time in seconds.

$$\theta_e = \int_0^t we \cdot dt \tag{18}$$

After the use of some calculus identities and a large amount of algebraic manipulation, the system of Figure 7 can be represented by the following ten state equations.

$$\frac{d}{dt} i_{Fq} = -we \cdot i_{Fd} + \frac{V_{Qn}}{L_F} - \frac{V_{Fq}}{L_F} - \frac{i_{Fq} \cdot R_F}{L_F} \tag{19}$$

$$\frac{d}{dt} V_{Fq} = -we \cdot V_{Fd} + \frac{i_{Fq}}{C_F} - \frac{i_{2q}}{C_F} \tag{20}$$

$$\frac{d}{dt} i_{2q} = -we \cdot i_{2d} + \frac{V_{Fq}}{L_2} - \frac{V_{Lq}}{L_2} - \frac{i_{2q} R}{L_2} \tag{21}$$

$$\frac{d}{dt} V_{Lq} = -we \cdot V_{Ld} + \frac{i_{2q}}{C_L} - \frac{i_{Lq}}{C_L} \tag{22}$$

$$\frac{d}{dt} i_{Lq} = -we \cdot i_{Ld} + \frac{V_{Lq}}{L_L} - \frac{i_{Lq} \cdot R_L}{L_L} \tag{23}$$

$$\frac{d}{dt} i_{Fd} = we \cdot i_{Fq} + \frac{V_{Dn}}{L_F} - \frac{V_{Fd}}{L_F} - \frac{i_{Fd} \cdot R_F}{L_F} \tag{24}$$

$$\frac{d}{dt} V_{Fd} = we \cdot V_{Fq} + \frac{i_{Fd}}{C_F} - \frac{i_{2d}}{C_F} \tag{25}$$

$$\frac{d}{dt} V_{Ld} = we \cdot V_{Lq} + \frac{i_{2d}}{C_L} - \frac{i_{Ld}}{C_L} \tag{26}$$

$$\frac{d}{dt} i_{Ld} = we \cdot i_{Lq} + \frac{V_{Ld}}{L_L} - \frac{i_{Ld} \cdot R_L}{L_L} \tag{27}$$

The 'e' superscript, designating the synchronous reference frame, has been suppressed to for convenience. The d and q subscripts will enable the reader to

distinguish a-b-c and synchronous reference frame representations. In comparison to the original state equations, these equations have noticeable differences. The most obvious difference introduced by the transformation is the appearance of ten state variables that incorporate the states of all three phases. This condition results from the matrix multiplication of the transformation and the assumption of balanced loading which eliminates the 'o' states.

Another change from the previous system equations is the appearance of *we* in each equation. This frequency is the reference for the q-d transformation. It must be stressed that nothing has changed in the system, only the representation of the system has been changed.

One drawback to this transformation is that the separation into q and d components complicates matters by making the problem multivariable with two inputs, V_{Qn} and V_{Dn}, and two outputs V_{Lq} and V_{Ld}. Another drawback to the q-d method is that DSP techniques must be incorporated to accomplish the forward and reverse transformations. That is, if the control is to be performed in the Synchronous Reference Frame (SRF), sensed quantities must be mapped into the SRF and control signals must be mapped out to the SRF and applied to the hardware.

3. Control in the Synchronous Reference Frame

With the system transformed into a more convenient representation, the controller can be developed. The control methodology chosen was a two-loop system consisting of a fast inner current control loop and a slower outer voltage control loop. This type of control can give excellent results even compared to more "exotic" and "sophisticated" control techniques [10]. Step changes in the load impose a very rapid change in the current through the system. Capacitance in the system causes the voltage to change more gradually than the current. In the two-loop control system these faster current changes are fed forward to create an anticipatory control signal, speeding up the overall response of the controller.

The number or current and voltage sensors available for use as inputs to the controller are limited to minimize cable runs to the Ship Service Inverter Module and, as a consequence, improve survivability. Thus, the controller was designed to use only parameters physically measurable local to the inverter. These quantities include the input DC voltage and current, the filter inductor current, the filter capacitor voltage and current, and the line current out of the unit. Note that the phase voltage is derived from the switch positions and the input DC voltage as discussed in Chapter II. The phase voltage must be derived since the voltage at the common neutral is not measured. An even more significant parameter missing from these measurable values is the load voltage and current. The plan is to use the inverter filter's voltage and output current as a reasonable approximation to the load current and voltage. If the filter capacitor current can be accurately commanded, regardless of the load current, then the output voltage can be precisely regulated as well since it is merely the integrated and scaled value of the capacitor current [3]. The validity of this assumption is shown in the results section of this chapter.

The commanded values for the controller are given by V_{Qn}* and V_{Dn}*. Following the design methodology in [10] using the measurable quantities, the controller is specified as detailed in Figure 8.

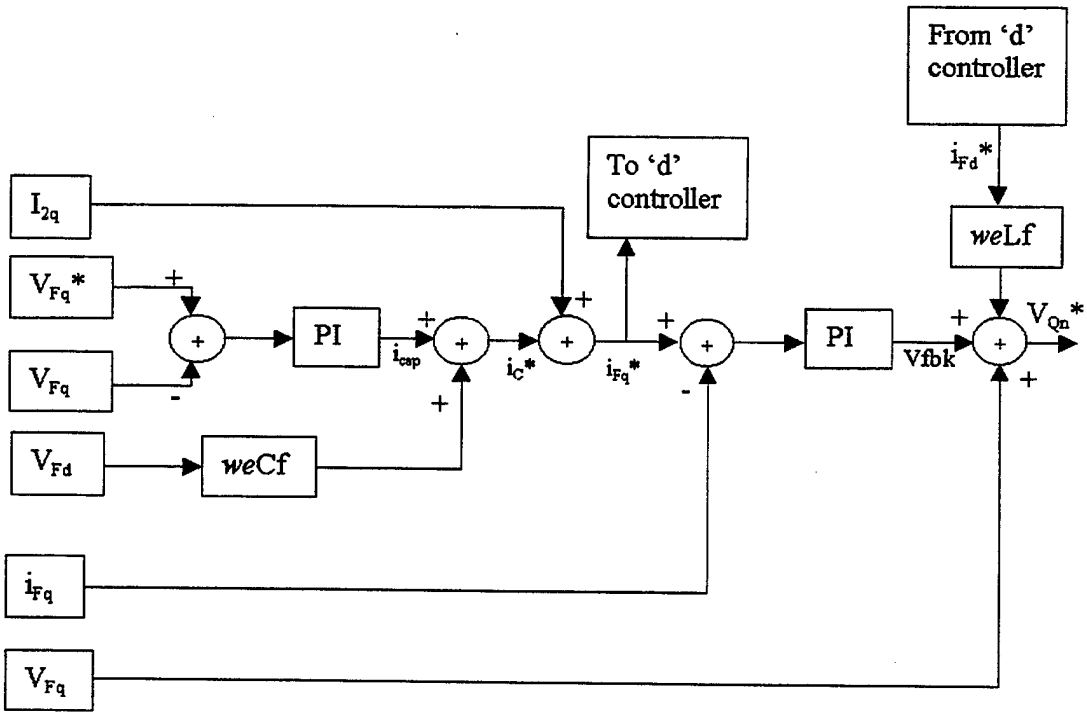


Figure 8. The Q Portion of the Controller for the Stand-Alone Inverter.

Careful inspection of the controller reveals some interesting characteristics. Although this is the controller for the q variables, two d terms i_{Fd}* and V_{Fd} appear. This phenomenon occurs since the q and d variables are intimately related as was shown in the system model of the inverter (the q and d dynamics are coupled). Additionally since both q and d have two PI controllers, four more states are added to the overall system.

Recall that the load parameters are indirectly maintained by controlling the filter parameters. Referring to Figure 8, V_{Fq}* (the commanded filter voltage) is compared to the actual filter voltage and sent through a proportional and integral controller forming i_{cap}. The feedforward term, representing the desired steady-state filter capacitor current $V_{Fd}weC_f$, is added to i_{cap} to form i_C*. Subsequently, the desired filter inductor current i_{Fq}* and the line current i_{2q} are added to the capacitor current. The i_{Fq}* term is used in both the q and d controllers (as is the i_{Fd}* term). This desired filter current is compared to the actual filter current and is sent through another PI controller creating the voltage feedback term V_{fbk}. This feedback term is then added to the measured load voltage (in this case

approximated by the filter voltage). The commanded filter current feedforward term is added to this sum giving the commanded switching signal V_{Qn}*. V_{Qn}* can be converted into an a-b-c sine-like quantity and compared to a triangle waveform to generate switching signals. The conversion to a-b-c values for balanced loads is given below in Equations (28) - (30).

$$v_a = V_{Qn} \cdot \cos(\theta_e) + V_{Dn} \cdot \sin(\theta_e) \tag{28}$$

$$v_b = V_{Qn} \cdot \cos(\theta_e - \frac{2\pi}{3}) + V_{Dn} \cdot \sin(\theta_e - \frac{2\pi}{3}) \tag{29}$$

$$v_c = -v_a - v_b \tag{30}$$

Only the q portion of the controller is shown, but using the same reasoning and use of similar system equations, the d portion can be derived. The d controller looks exactly the same as the q controller with two exceptions. The d and q subscripts are swapped indicating the variable change, and the *we* terms are negative (a result of the transformation).

With two PI controllers for both q and d variables, and two variables per PI controller, there are a total of eight gains that must be determined. Trial and error guessing of the gains is a very inefficient strategy for assigning the gains and effecting an acceptable transient response. Script files for Matlab are discussed in [10] which use pole placement and a Newton-Raphson algorithm to find the controller gains. The script files were modified for the proposed design and the gains were calculated in Matlab. The resulting system gains are shown in Table III-2.

k_{pcq}	3.16E-1
k_{icq}	5.06E2
k_{pcd}	3.59E-1
k_{icd}	5.06E2
k_{pvq}	9.26E0
k_{ivq}	4.05E3
k_{pvd}	1.77E1
k_{ivd}	4.05E3

Table III-2. Controller Gains for a Stand-Alone Inverter.

The subscript coding on the gains is as follows. The q and d represent the controller the gain is used in, and the p and i refer to the proportional and integral gains of the PI controller. The v and c refer to the type of input to the PI controller. The v gains apply to the first PI controller where the input is a voltage difference, and the c gains apply to the second PI controller where the input is a current difference. With these gains calculated, the inverter and its controller can be tested in Matlab and ACSL.

D. RESULTS

The model was run with the parameters determined in the previous sections. To simplify the problem further, the commanded d voltage was set to zero, and the commanded q voltage was set to the desired peak phase voltage in the a-b-c frame.

1. The Integrated Model

The first test of the inverter was to judge the validity of controlling the filter voltage instead of the load voltage. The first series of tests were conducted to determine the effects of changing loads on the inverter's ability to maintain the bus voltage. The second series of experiments determined the impact of varying line impedances on inverter voltage control.

The nominal values used in the simulation were: L_F=5.63e-5 H, R_F=0.1 Ω, C_F=1.123e-2 F, R_2=0.3 Ω, L_2=1e-6 H, R_L=20 Ω, L_L=0.017 H, C_L=1e-5 F. Arbitrarily the commanded voltage was set to 400 volts with the DC supply voltage at 950 volts. The peak values and the sinusoidal quantities were recorded for all of the model parameters. Plotting the peak values aided in analysis of the data, while the sinusoidal values revealed the quality of the AC waveform. The parameters of interest were the load voltage and the load current. The model was run with the nominal values giving a peak load voltage of 394.7 volts and a peak load current of 18.8 amps.

The first test demonstrated the dependence of the controller on the load resistance. Phase A's load voltage was calculated from the peak d and q state variables using Equation (31).

$$V_{Load,A} = V_{Load,q} \cdot \cos(\theta_e) + V_{Load,d} \cdot \sin(\theta_e) \tag{31}$$

This test demonstrated the natural tendency of the inverter to respond rapidly to load changes. The load resistance was step-increased to 1000 Ω at time t=0.02 seconds. As shown in Figure 9, this change was not even noticeable on the sinusoidal plot of the Phase A load voltage, but is very prominent on the Phase A current plot. The response of the inverter was extremely fast.

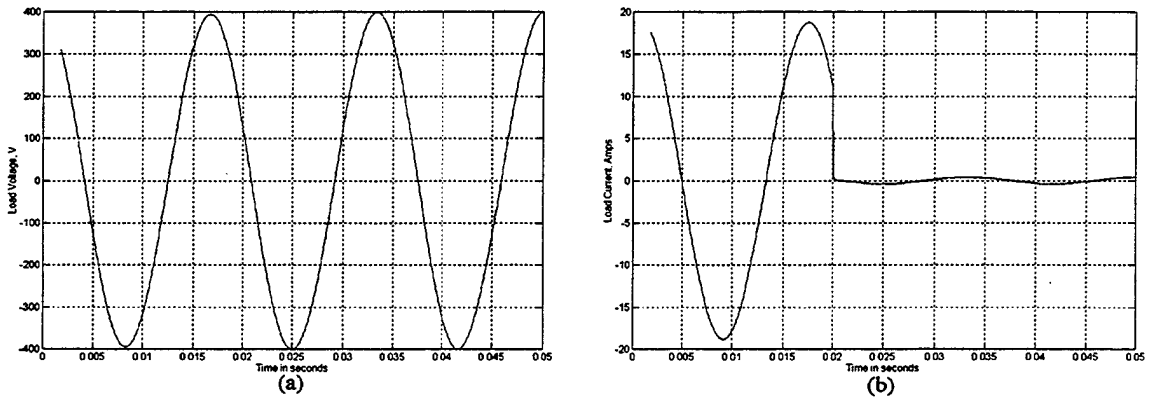


Figure 9. The Load Voltage (a), and the Load Current (b) for a Resistive Load Increase from 10 to 1000 Ω.

The peak value plots are shown in Figure 10. These plots offer increased utility since their scales can be smaller, and the system's transient behavior is not obscured by its cyclical nature. Figure 10 illustrates the response time of the system. For the change in real power loading, the voltage reaches its new equilibrium value in approximately 3 ms, while the current response through the load is almost instantaneous. Additional testing was conducted for many values of load resistance, all gave similar results. Due to the small scale of the voltage magnitude plot, some harmonics are present on the instantaneous peak waveforms.

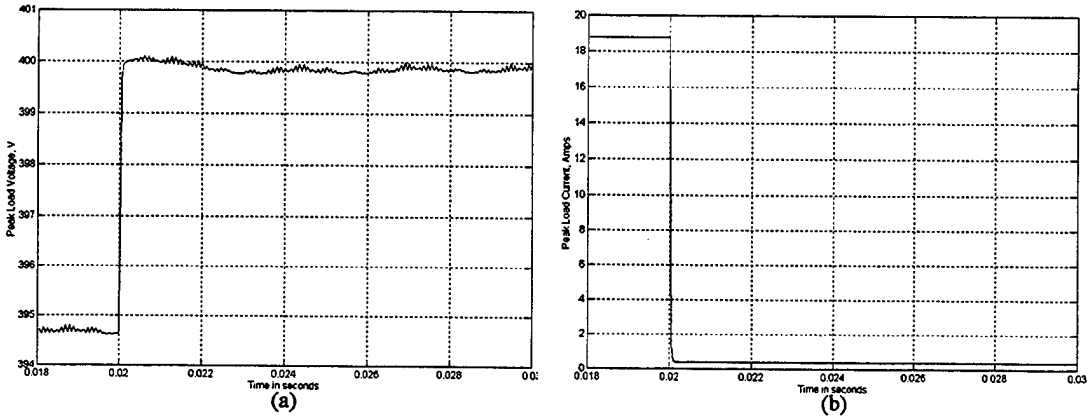


Figure 10. The Peak Load Voltage (a), and Load Current (b) for a Resistive Load Increase from 10 to 1000 Ω.

The resistor is only one component of the model's load. Effects from changes in the load capacitance and inductance were also examined. The inductance of the load was increased six fold, but this only caused a 0.05% increase in the real load voltage. The load inductance was then reduced by two orders of magnitude causing a 0.10% decrease in the real load voltage. As expected, changing the size of the inductor or capacitor at the load does not significantly effect the amount of real power drawn by the load. The only significant effect seen from raising the load inductance was the longer response time of the system. The system stabilized at the new values of voltage and current 10 ms (three times slower) after the transient as seen in Figure 11.

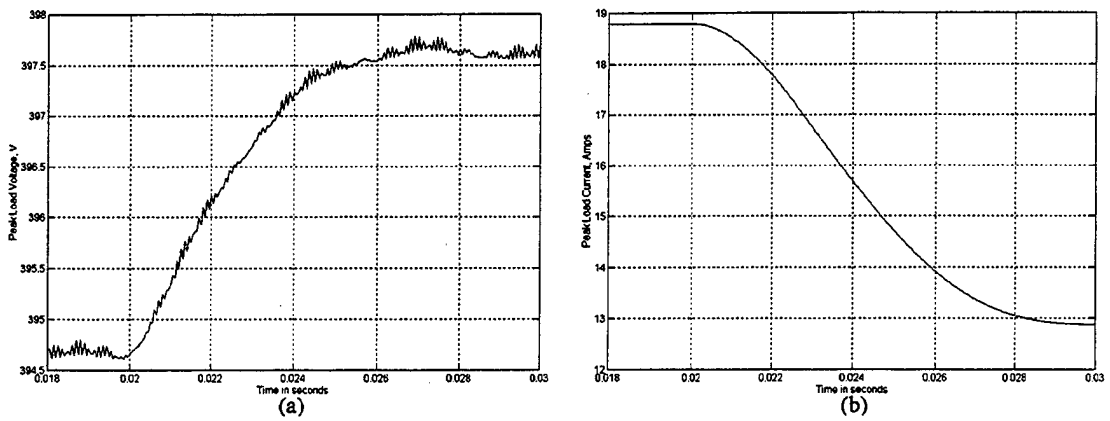


Figure 11. Response of Peak Load Voltage (a), and Peak Load Current (b), for a Load Inductance Increase from 0.017 H to 0.06 H.

Changes in the load capacitor also exhibited insignificant effects on the load voltages. Additionally, slight changes in system response time for changes in load capacitance were also observed.

The second set of tests aimed at confirming the validity of local voltage control evaluated the effects of line impedance on load voltage control. Changes to the line impedance produced expected results. The highest line impedance expected on a single inverter, at full power (19 peak amps), was 300 mΩ and 1 μH. Lowering the line resistance caused the commanded filter voltage to more closely match the desire load voltage. Lowering the line inductance sped up the system response. If the line inductance is set in the milli-henry range, voltage control at the load is lost.

Thus, for a wide range of expected and unlikely system parameters, the inverter model and controller are stable and very robust. The model and controller worked well to compensate for system modifications. Next, the effects of large changes in commanded voltages were investigated.

The effect of changes in the command voltage tests the stability and responsiveness of the controller. For these tests the nominal values were used for the components of the model. The first set of tests looked at the transient response of the controller to commands. The second set of tests examined the effects on inverter real and

reactive power output for changes in the commanded q and d voltages. In Figure 12 the commanded q-voltage for the filter is stepped from 400 to 300 volts at time t=0.02 seconds. The load voltage stabilizes at its new value of 396 volts in approximately 10 ms with a 10% undershoot. Voltage command changes from 400 to 475 volts had approximately a 10 ms stabilization time with a 10% overshoot. For commanded voltages greater than $V_{dc}/2$ (the upper bound for linear modulation range), the output was not stable due to a decrease in gain when the system enters the overmodulation region.

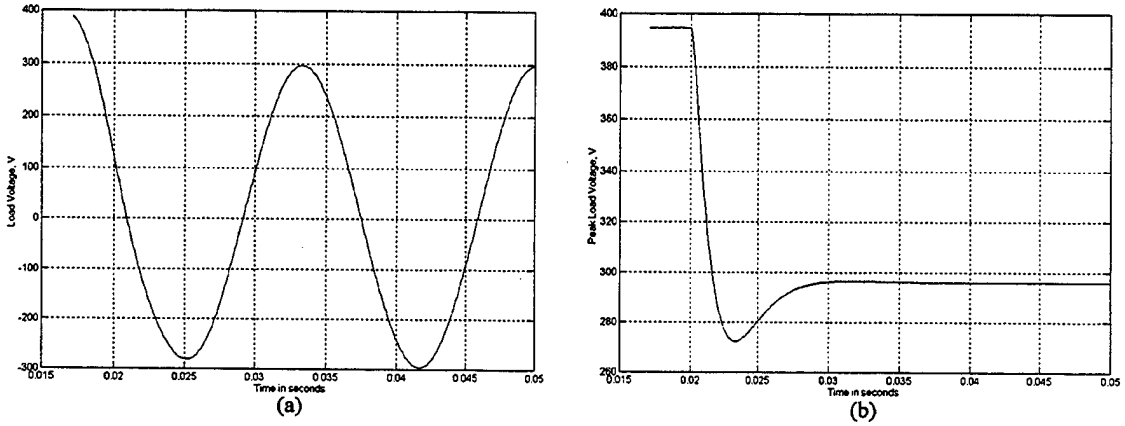


Figure 12. Effect on the Load Voltage (a) and Peak Load Voltage (b) for a Step Change in Commanded Filter Voltage from 400 Volts to 300 Volts.

Effects of commanded voltage changes on inverter power output were also examined. An increase in the commanded q-voltage of five volts caused the real power output of the inverter to increase by about three percent. This is not very significant.

Changes in the commanded d-voltages introduce a slight phase shift in the inverter voltage changing the reactive power delivery. Increasing the value the commanded d-voltage from its nominal value of zero to three volts impacted power so slightly that it was barely discernable from the noise. Even increases of 40 volts caused reactive power to change by only one percent for the single inverter.

Normally, a navy ship's bus will remain a constant fixed value of voltage so the dynamics of the system for changes in commanded voltage are not of great concern for the individual inverter. The importance of the controller dynamics becomes critical if the

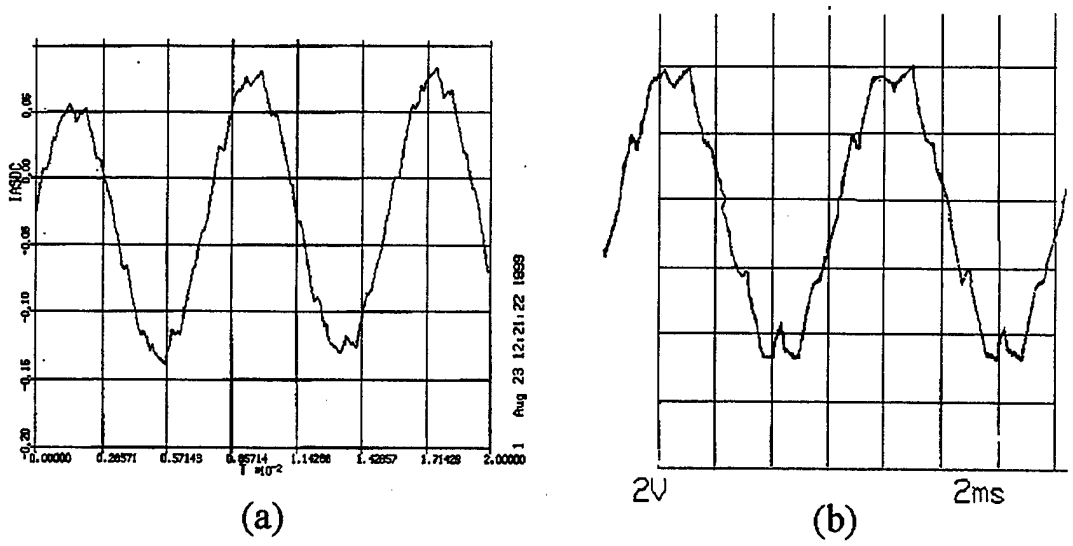
current or power output of the inverter needs to be controlled. If an inverter was attached to a utility-powered bus, varying its voltage output relative to the utility bus could control its current output. Thus, good voltage control dynamics determine how well an inverter can share power with another source.

2. Agreement with Experimentation

The previous results are only meaningful if they represent 'real-world' results. The modeling validity of ACSL was tested against the Lab-Volt inverter and open-loop control even before the inverter was modeled in the synchronous reference frame. An open-loop three-phase inverter with a simple RL load was constructed in the lab and modeled in ACSL. Open-loop operation is simple to model because there are no AC feedback quantities to deal with. The one problem encountered was that the output filter on the inverter is internal to the unit, and the equipment manual does not list the values. So the modeled values for the output inductance and capacitance were measured estimates of the real values. Additionally, the sine-triangle modulation is internal to the controller so the frequencies used had to be estimated.

Upon "tweaking " the values, the results of the model very nearly matched the values from the Lab-Volt hardware as seen in the two waveforms of Figure 13. The axes on the graphs are slightly off but the values for voltages and the general waveform shape are in agreement.

Although many assumptions concerning the hardware were required, the results prove that ACSL can be made to mimic the real world with very good precision.



(a) (b)

Figure 13. The ACSL Generated Waveform (a), and the Hardware Generated Waveform (b).

IV. SHARING METHOD FOR MULTIPLE INVERTERS

With a working inverter model available, the problems associated with paralleling inverters can now be addressed.

A. PARALLED INVERTER DESIGN ISSUES

This section deals with the design issues surrounding the paralleling of three-phase voltage controlled inverters. Issues concerning the sharing of the load between the inverters and the coupling of inverters are addressed. Also, the modifications to the controller required for load sharing is discussed, and the topologies for inverter control are explained.

1. Sharing the load

The first design issue to understand is how AC generating/conversion machines operate in parallel on a bus. Paralleled inverters producing the same peak output voltage to a bus can be supplying very different amounts of current to that bus. Inverter load sharing is determined by the voltage difference and the phase difference between the inverters [2]. The differences in line resistances between the inverters and the bus influences their current sharing. Since the real power out of an inverter is proportional to the product of the current and the voltage, if the currents are unequal the sharing of the true power is unequal. Some of the inverters could be carrying little load, while others could be overloaded. Fast, low output impedance inverters tend to be overloaded [2]. To prevent an unnecessary overloading of an inverter, the impedances of all the units must be identical, or a control scheme must be used to correct for the impedance differences. If the inverter internal impedance can be made large, both the control and the protection of the units becomes easier [2]. Therefore to protect against overloading, the impedances of the inverter should be high, but the design is also constrained by the response of the output filter to the desired fundamental frequency. Furthermore, this

impedance solution still does not cause the load to be equally shared, it just adds some semblance of overload protection. Consequently, an additional current control algorithm must be added to the inverter's control system to equalize the inverter loading.

Even if paralleled inverters have similar impedances, small system perturbations may cause them to 'fight' over their current share. This fighting may lead to large oscillations in the inverter currents. If the current oscillations are unstable, they will grow until the inverter power limit is exceeded or the current limits on the hardware are exceeded. This can happen since the inverters can act as a load or a source thereby absorbing the additional current supplied by the other inverter(s). Just as in overloading, to prevent these oscillations from occurring, a current control method between the inverters must be developed.

2. Coupling Concerns and Startup

How the inverters react to each other's presence on the bus must be minimized. In the best case, adding an inverter to the bus should only minimize the current requirements from each inverter. Once again, the output filter of the inverters is of major concern. The filter must pass the fundamental, block the harmonics and switching noise, and not interact adversely with other filters on the bus. As discussed in [5], oscillations can occur as the result of filter interactions between paralleled units. However, a non-interactive filter may not be as effective at harmonic elimination. Also, the filter must still allow for the inverter to process power flowing into or out of the output. Chandorkar in [5] advocates connecting paralleled inverters using a tie bus having a resistor and inductor. This solution seems to work, but it severely restricts the modularity and flexibility of the overall design. In [2] Kawabata stresses that the filter capacitors of each inverter and the cable inductance between inverters form a resonant circuit. But he maintains that this resonance may be avoided with suitable selection of series output inductors.

Interactions between the units under normal operating conditions are a concern, but so is the interaction under failure conditions. An important implementation problem to be addressed is how the paralleled units will behave after a catastrophic failure of a unit. In the best case, the other units will pick up the load and hopefully not be overloaded. In the worst case the compromised unit fails due to an internal short of a phase or between phases. The failed inverter now becomes an enormous load on the system, which might take down the other healthy inverters. This thesis will not consider the design of fault protection, but for the actual implementation a protection mechanism must be in place to isolate failed units quickly.

In addition to isolation problems, starting the system or returning an idle inverter to service is a delicate task for an AC power system. Unlike paralleling DC-to-DC converters, PWM inverters require the bus voltage, frequency, phase, and sequence to be matched in order to parallel units. Initially the inverters are isolated from each other probably with a set of breakers. Before those breakers are shut, the incoming inverter must meet at least three criteria. First, its voltage output (peak or rms) must match the buss' voltage. If the voltages don't match you will have a ΔV across the low resistance contacts of the breakers. The resulting high surge current could damage the breakers or cause excessive transients on the AC bus.

The second requirement for the incoming inverter is that it must be in phase with the bus. This requirement is similar to the voltage requirement. If the bus and incoming inverter voltages are not in phase, the breaker contacts will again see a ΔV, with the same damaging consequences as before. The requirement for the units to be in phase is advantageous for synchronous reference frame control since knowledge of the filter characteristics and the measured phase at the filter enables the calculation of a system θ_e.

The last requirement is that the incoming inverter's output frequency exactly matches the bus frequency and that the sequence of the switch actuation is matched. Normally, the frequency is important for an AC generator power loading, but for the solid-state machine this is not so. This anomaly is discussed in future sections. With

the frequency, phase, sequence, and voltage magnitude matched, breakers isolating the inverter can be safely operated.

3. Current and Voltage Control

From the preceding sections it can be deduced that the control algorithm for a single inverter will not work for a group of parallel inverters. Maintaining the filter capacitor voltage is not sufficient to facilitate real and reactive load distribution between inverter units. But, the individual inverter control is very fast and stable and it seems reasonable to simply modify it to facilitate paralleling.

As stated in Chapter II, only two parameters of a PWM inverter are normally controlled: the output frequency and the output voltage magnitude. Additionally, the phase and sequence of the inverter output can be manipulated if desired. The controller modification must therefore manifest itself as a new loop to be added to the voltage control of the inverter. Where this loop gets its input is dependent on the topology chosen for the inverter system.

4. Inverter Topology

Determining the control topology for the parallel inverter system is very important. This topology is the starting point from which sharing design decisions will be based. In general the literature identifies two primary topologies, centralized and distributed. The definitions of centralized and distributed inverter systems are not standardized. In this thesis a distributed topology is defined as one in which the paralleled inverters do not require external controls or devices to operate. They can share information actively or passively, but loss of this information cannot cause the individual inverter to fail. A centralized topology is one in which there is a required dependence by each unit on an external device or control signal.

A centralized scheme relies on a single control unit to distribute data to the system network. A common computer could be used to take the power measurements from the

individual inverters, determine the average power for the system, then send feedback signals to the inverters to force them to evenly share the power load. If the central computer was powerful enough, the entire controller for all the inverters could be housed in this single unit. This arrangement would probably be more cost effective than having a controller at each unit. The direct switching signals would be output from the controller and sent to the inverters. If the inverters are physically far apart, there may be problems with signal timing and signal strength.

A less-centralized topology is the master-slave setup. Instead of the separate unit synchronizing the inverters, each inverter has its own local control unit with one of the inverters in the system designated the central controller. A popular use of this topology is for the master to be used to maintain the bus voltage while the other paralleled inverters are commanded to share the current.

Centralized control has three main advantages. First, the control can be fast and accurate since a single device coordinates it. Sharing is simplified by the common data being made directly available for processing.

A second advantage to centralized control is that is provides a single monitoring and control station. Measurements and maintenance of the control system is simplified. Troubleshooting of system problems will also be less complicated. If the system control is not operating correctly the problem can be isolated more quickly.

Finally, the issues of fault isolation and startup would be simpler to manage in a centralized system. For fault isolation, an external monitor can detect a malfunctioning unit and remove it from service before protective action from overcurrent or overvoltage must be used. For startup, the central unit could coordinate the incoming unit's parameters automatically making the startup sequence faster.

The are two key disadvantages to the centralized control topology. The most significant disadvantage is that there is a single point failure for the system. If the controller fails due to a bad part or through battle damage, the whole system goes down. This loss is unacceptable. Of course a standby controller could be activated upon failure of the main controller, but this will not solve the problem completely. The reason for this

is that the centralized control scheme also requires direct wiring from the units to the controller. Hence, damage to this communication line will also render at least a portion of the system unusable.

The communication wiring may require conduits, bulkhead penetrations, ship alterations, etc. So the second disadvantage to the centralized scheme is that it limits the system's flexibility for future growth and flexibility during initial installation. Installing an additional unit in the future may require extensive work just to route the communication wiring. If more than one extra unit is required in the future, the master may not be designed to handle the extra inverters required and it too might need to be replaced causing the upgrade cost to skyrocket. To limit this wiring hassle, the master control and the inverters might be required to all be in the same space. This option limits the design flexibility and makes the whole system susceptible to destruction from a single casualty like flooding.

In contrast to the centralized control is a distributed control. In this topology each inverter has its own sensing equipment and control equipment. Because of this autonomy, the inverters can be located in different spaces throughout the ship if desired. The only two requirements on the inverter location are that it is in a space that has access to an SSCM DC source and the AC bus. Distributed inverters share only a minimal amount of data if any at all. This data is not necessary for the operation of an individual unit, but is needed only for load balancing. In order to communicate this data without the necessity of hard wiring the units, two techniques could be used. The most obvious method is the use of transmitters and receivers. This 'wireless' solution would use an available frequency band to allow the inverters to trade loading data. The second method of communication available to distributed inverters is the use of the power bus itself. Small signals could piggyback on the fundamental frequency. These signals could carry the data necessary to allow the inverters to share the bus loads.

The advantages of the distributed control topology address the disadvantages of the centralized control topology. For distributed control there is no single point failure. Failure of communication might lead to system inefficiency but not system failure. This

fact is very important. A distributed system is much more likely to survive battle damage or casualty. The reliability of this configuration in a hostile environment is extremely desirable. Likewise, the communication will not require hard-wired connections, so the problems addressed in the centralized section with installing the initial system and updating it later are eliminated.

The disadvantages of the distributed system are the opposite of the centralized advantages. The distributed system is harder to control and troubleshoot than the centralized system. Individual inverters must have a self-diagnostic feature to shut down when local faults are detected. Finally, the control signals used (if any) will probably be available at a reduced speed and reliability.

B. DISTRIBUTED COMMUNICATION OF INFORMATION

Distributed communication is the major hurdle to developing a true distributed system. Careful analysis of the requirements and options for this communication is presented.

1. Limited Data

The amount of data required by other inverters in a distributed environment is minimal. There is no need for the inverters to send large amounts of data back and forth reporting on their present status. Perreault in [6] agrees that in order to implement a distributed load-sharing scheme, only a very limited amount of information needs to be shared among the individual units. So, the one question that had to be answered is what is the minimum amount of data that needs to be transmitted to ensure that the inverters share the real system load. The answer is only one piece, the RMS load current (or average power of the load). If this signal can be generated and sent to all the inverters, then they can simply compare their own output to the average then lower or raise their output as necessary to attain the system average. The problem is that for the distributed

system there is no distant wiring or sensors permitted. The question now becomes, how is this quantity generated?

2. Completely Autonomous

One method to solve the data problem is to design a system that requires no data transmission at all. This type of system uses only local information to gather system knowledge. The trick is to incorporate a system-level response at the local level then use this information to gather system information indirectly.

One such method used in generators is droop control. In this method, the paralleled generators or inverters are designed to respond to increases in real load with a programmed decrease in system frequency. The slope of the frequency drop is inversely proportional to the power rating of the generator relative to the other generators on the same bus. For example, the slope of a Generator A with half the real power rating of Generator B will have twice as steep a frequency drop. As stressed in [5], if the droop slopes of every unit powering the system are chosen such that the product of the rated power of the unit and the droop slope is the same for each unit, then the units will take up load changes in proportion to their power rating. This method can also be applied to the reactive power load sharing by causing a voltage drop as the reactive load increases.

To illustrate droop load sharing as it applies to AC generators, a real load will be shared between two generators 'A' and 'B' of unequal power rating (see Figure 14). Loading figures of this type are referred to as House Curves. In this case the loading on the operating generator will decrease. Figure 14.a shows a single generator 'A' at full load operating with a frequency of f_1. The load line represents the real power required by the load. A second generator 'B' is started in parallel with 'A'. Figure 14.b shows how this effects the system.

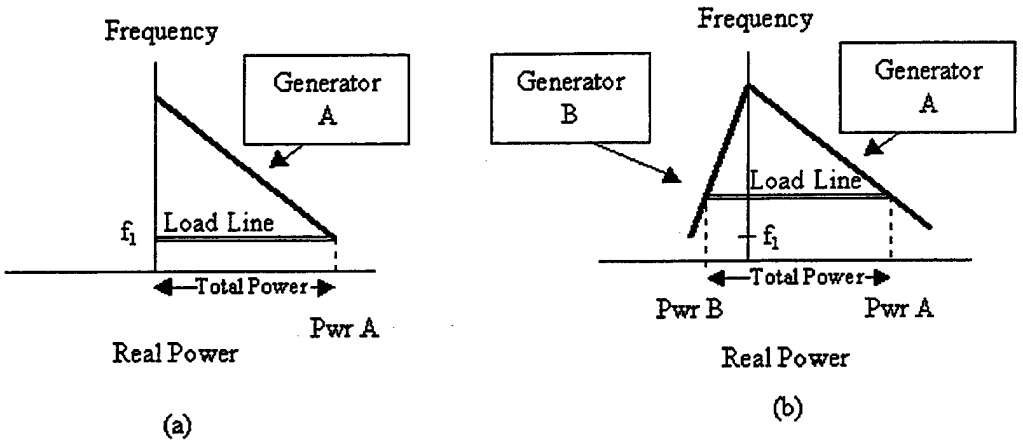


Figure 14. House Curves for (a) Single Unit A, (b) Units A and B Paralleled.

The real power required by the load remains the same, but the system frequency changes. As A's load is reduced, A responds by raising the system frequency as determined by its droop slope. In steady state due to the programmed droop of each generator, the units will share the load in exact proportion to their load handling capability (i.e. both at 75 % capacity).

The droop technique has advantages and disadvantages. A major advantage to this technique is that the technique is easy to understand. Load sharing using voltage and frequency droop has a merit of no common parts and wiring between the units [2]. A modification to the control system could be used to measure the real power out of the generator and shift the generator's output frequency accordingly.

A further advantage of droop control is that it is a mature and well-understood method of parallel generator operation. Droop control is commonly used in Naval Electrical Generation Systems as a low-level sharing mechanism. Prevention of overloading in conventional power systems is achieved by introducing a droop in the frequency of each generator as real power increases [5]. Typically operator action is required to correct the system frequency after load changes. Requirement of operator action is not a large concern in this case since the generators normally used in Naval ships

have such a large load capacity that only a major change in loading on the bus will cause significant frequency shifts.

Although having to restore the bus frequency for large generators is not a problem, it is a significant disadvantage to droop in smaller AC generators. These smaller generators or inverters have a much steeper droop slope. Compared to a single large generator, for a large load change, a single smaller generator will have a much larger bus frequency change. A significant change in bus frequency can cause some equipment to function improperly.

Correcting the bus frequency automatically can also be a problem. Assuming the desired frequency f_1 is 60 Hz, both units are required to shift their entire load curves until the system load line lies at f_1. The coordination of this curve shift is critical for proper system operation. If one unit shifts faster than the other, or if the shifting controller is not overdamped, then serious oscillations can occur. Ensuring that the controller is overdamped slows its response time. Even with overdamped controllers, the droop approach tends to fail when more than two identical units are paralleled using this mechanism. Small system perturbations lead to 'fighting' between the units as to their proper portion of the load line. The oscillations produced can cause system failure or unacceptable frequency swings that may damage sensitive equipment. For this reason, the current naval policy is to not allow the operation of three AC sources in parallel on a bus.

Many authors recognize the limitation of droop control, but they also remain loyal to its implementation. A way to minimize droop's disadvantages is to use it in combination with another control scheme. One such hybrid method is called Controlled Restoration. Controlled restoration combines droop control with another controlled parameter. In [4] frequency is the parameter whose error restoration is controlled. By making the rate of frequency restoration a common constant for all inverters connected to the system, the inverters will share the load evenly [4]. The rate at which the curves are shifted up or down is proportional to the power rating of the inverter multiplied by its instantaneous frequency error (see Equation 32). In this Equation P_{oi} is the top of the

droop line touching the frequency axis, K is the frequency restoration constant, P_{iR} is the inverter's power rating, and $\Delta\omega_i$ is the frequency error.

$$\frac{d}{dt}P_{oi} = K \cdot P_{iR} \cdot \Delta\omega_i \text{ [4]} \tag{32}$$

To coordinate the restoration of the frequency each unit has a built in clock to synchronize the restoration processes. This solution looks promising, but once again we are manipulating system frequency which complicates the system control.

Use of this varying bus frequency to distribute data to inverters in a system has some drawbacks. Efficient control algorithms using the synchronous reference frame will no longer have a constant θ_e value for the transformation. This fact complicates the control by requiring a fast frequency detection circuit for use in the transformation. Not only will the control slow down, but also the dynamics of the detection circuitry may make the controller output less than satisfactory.

The biggest potential flaw to the frequency droop technique for inverters is its implementation. Frequency techniques work great for generators with large magnetic fields locked to the system frequency, but what about inverters? What prevents one inverter from operating at 60 Hz while the other operates at 61 Hz? This question will be explored more fully in later sections.

With droop and modifications of droop seemingly the only available options for a fully autonomous system, other communication methods were investigated and are reported on in the next sections.

3. Shared Information Methods

There are many non-droop related sharing methods available in the literature [2]. Most of these methods are designed using a centralized control topology that would require them to be modified to work in a distributed network.

The two media for distributed communications discussed previously were wireless and power line transmissions. Onboard ship the wireless solution is not

appropriate. Heavy interference from nearby equipment or bulkheads may attenuate the airborne signals. Even worse, the inverter signals could interfere with sensitive shipboard systems. If the inverter systems were close to any magazines or weapons such as torpedoes, then the transmission strength and frequency would be severely limited for safety reasons. This leaves power line communications as the only other alternative.

One such power line communication technology is the Consumer Electronics Bus (CEBus). This system uses spread spectrum technology to enhance the signal to noise ratio of the data communicated over power lines. In October of 1998, Northern Telcom Ltd. demonstrated the ability to transmit data at speeds in excess of one megabit per second over commercial power lines. The signal used for CEBus is a swept-frequency pulse-chirp system. One advantage to this type of technology is that it is a tested technology with significant data transfer ability. Additionally, since the transformations required for the controller need DSP equipment anyway, perhaps the CEBus hardware could also be modified to perform the controller tasks. One substantial drawback is that this system currently uses frequencies much higher than the system frequency to transmit its signals (~143 kHz). This frequency would be severely attenuated by the inverter's output filter.

Another drawback to advanced power line networking approaches such as CEBus is that they are designed for systems which must communicate large quantities of data usually in a full duplex mode. The CEBus is an overkill for the limited data that needs to be shared in one portion of the proposed DCZED. Additionally, networking problems such as dynamically adding and removing hosts from the system are inherent to a full-scale networking solution such as CEBus.

An alternative to a full-blown network solution is to develop a system that transmits only the data necessary. In [6], Perreault suggests the use of an injected signal onto the power grid to communicate information between buck chopper DC-to-DC conversion units. Moore tested this idea in [11] for DC-to-DC converters with positive results. It may be possible to expand this idea to DC-to-AC inverters as well.

The operational mechanics of the inverter provide a method to generate this communication signal without a separate generation device. Signal injection uses perturbations intentionally added to the control sinusoid of the PWM switching scheme to pass information via the power bus. It is a distributed method requiring no communication connections between the inverters since the signal flows out of the inverter on its power connections.

This injected signal must meet three criterions. First, it must be detectable by the other inverter units. This requirement is extremely difficult to implement because of the required output filter in the system due to switching noise and harmonics. As discussed previously, this filter must have a flat response at the fundamental frequency, and must maximally attenuate the switching harmonics and noise. Secondly, the signal must be restricted to a band between the fundamental and switching frequencies. The signal produced must be significantly lower in frequency than the switching frequency to ensure it is not severely attenuated. Additionally, the signal frequency must be higher than the fundamental frequency to minimize its detrimental effects on motors and other loads. The last criterion is that the signals must contain the required information, and this information must be extractable in a timely manner.

The goal is to share power without interconnections. To compute its share of the power load, each inverter must know two pieces of information either directly or indirectly. First, the inverter must know what the total power or total RMS current to the load is. Second, the inverter must know the aggregate number of units connected to the bus. As stated in a previous section, the minimum amount of data required is a single quantity, either the average power of the load or the average RMS current to the load. The inverter with this averaged information can then determine if it must acquire more load, or if it must relinquish some load.

The next issue to address is whether the data should be sent and received in digital form or analog form. Based on the filter design, the window of frequency ranges usable for data transmission range from about 600 to 800 Hz. The choice for a lower bound of 600 Hz was used to allow the signal's frequency to be at least one order of magnitude

larger than the fundamental frequency. The 800 Hz upper limit was chosen to minimize the effect of the system filter on the injected signal. The advantage of digital data is that it can use the noise reducing techniques of spread spectrum technology. The main disadvantages to digital data are that is requires extensive computational power and conversions between analog and digital forms of the data. Additionally, each inverter would need to be given non-overlapping frequency ranges to transmit in. Operators would have to know which frequency ranges were in use to ensure new units brought online would not conflict with these prior frequency band assignments.

The alternative data form is analog. Analog data can be encoded in the signal's amplitude, its frequency, or both. The filters and other components used in the inverter operation make the amplitude approach unreliable. Frequency encoding of data is a more robust encoding scheme for this environment. Frequency encoding is suggested in [6] as a means to enable all inverters on a system access to the average current output of the converters. The units transmit frequency signals proportional to their current output. These signals are available to all the converters on the system. The average frequency (current) can then be calculated using a filter, a differentiator, and a RMS calculating chip. This RMS method effectively converts the frequency information into a DC amplitude value proportional to the average power output of the converters.

Another analog method investigated, using frequency injection, was the use of beat frequencies from the injected sinewaves. It can be shown that added sinusoids, whose frequencies are relatively close, produce a beat pattern with a known carrier and modulating frequency [13]. The resultant carrier frequency is the difference between the two sinusoid's frequencies. The resultant modulating frequency is the average of the added sinusoid's frequencies. Experimental tests and Matlab simulations proved that the averaged frequency obtained from the modulating frequency also applies for multiple sinewaves if certain conditions were met. This well-defined pattern emerged if the transmission frequency range was chosen to be within +20% of the minimum frequency transmitted. If the minimum frequency was chosen to be 500 Hz, then the zero to one hundred percent range of frequencies would be 500 Hz to 600 Hz. Thus, many inverters

could all be injecting a frequency signal proportional to their power onto the bus, and the beat dynamics would generate a modulating frequency proportional to the average frequency (average power) of the connected units. This average power could then be compared by each inverter to its present power output, and adjustments made.

Since the transmitted waves would occupy the same frequency band, destructive interference could be avoided by ensuring that the first zero of the transmitted wave corresponded to the zero of the phase's voltage. Since all phases must be in synch, there is no chance for destructive interference.

Tuladhar discusses a different distributed communication method based on circulating current in [12]. Here again frequency is used to encode the data, but the extraction of the data is quite different. This method uses the fact that the injected frequency difference between two inverters manifests itself as a phase difference. The phase difference will cause a small real power to flow from the unit that injected the lower frequency signal to the unit that injected the higher frequency signal. Inverter's that detect this positive power flow recognize that they must reduce their power output. When the power sharing is complete, no inverter will receive this power signal.

4. Dealing With Noise

Many of the discussed signaling methods seem usable under ideal conditions, but what happens when real-world loads are used? The biggest factor to overcome is the inherent noise on an AC distribution system. The sharing method used must be resistant to the noise encountered in a typical Naval AC bus. None of the analog signals are truly immune to noise spikes if they fall in the communication range of interest. The range of 500 Hz to 600 Hz was arbitrarily arrived at so it may need to be revised after further analysis. A better method would be to analyze the spectrum of noise on a typical Naval AC distribution bus during steady-state operation and loading transients. From these observations, the frequency window with the lowest magnitude and number of noise spikes could be selected.

In contrast to the analog communication methods, the digital solutions are designed to use spread spectrum techniques to mitigate the effect of noise. Digital communication may be required if the noise environment is too hostile. To achieve desired noise levels, the bandwidth of the sharing signals is much larger than the analog case. The bandwidth will also be higher since each machine must be assigned a non-overlapping frequency window. Therefore, the price that must be paid in order to realize the high SNR of a digital system is an increased bandwidth requirement. Additionally, designing an inverter filter to make available the required bandwidth while still managing its other functions would be challenging.

V. PARALLELING INVERTERS

This thesis has been building up toward the ultimate goal of designing a distributed network of parallel inverters. In this chapter, model modifications required for paralleling inverters are investigated. Furthermore, the ability to share loads completely autonomously is explored. Finally, the effects of signal injection techniques are also considered.

A. MODEL MODIFICATIONS

As was planned, the single inverter model required little adjustment to be incorporated into a parallel network. The most notable change was shifting the line resistance R_2 from 0.3 Ω to 0.07 Ω. This change is justified in the next section. The inductance on the load was also increased from 0.017 H to 0.021 H to represent a more realistic 0.8 lagging power factor load. These changes caused a shift in the calculated eigenvalues of Chapter III, but they remained comfortably in the left-half plane. Another change in the single inverter model was the inclusion of a phase-locking mechanism for the inverters. A phase-locking mechanism prevents the inverters from drifting out of phase with each other and maintains the 60 Hz primary bus frequency. This phase locking is implied in the model and is not specifically detailed. The ACSL code found in Appendix A documents how this locking is accomplished.

Before attempting a voltage-droop sharing algorithm on paralleled inverters, the effects of drooping the commanded q and d voltages was tested on a single inverter. These were conducted with R_2 set to 0.3 Ω rather than the 0.07 Ω used in the final model. Simulation in ACSL demonstrates that the trends of interest remain the same for the different resistances only differing by a scaling factor. For a stand-alone unit Table V-1 documents the effects of changes in commanded q and d voltages on load voltage, real power, reactive power, and load current. Table V-1 illustrates that for an individual inverter a change in 5 V for V_{qcom} causes the V_{Lq} to change by 5 V and the real power to

increase. A change in V_{dcom} of 40 V has very little effect on real or reactive power. The reason this 40 V change in V_d has so little an effect on the inverter output is because the value of the V_q voltage dominates by an order of magnitude. Thus, the results are as expected.

R_2	V_{qcom}	V_{dcom}	V_{Lq}	V_{Ld}	R_{pwr}	Q_{pwr}	I_2
0.3	400	0	394	0.16	11800	-440	19.8
0.3	405	0	399	0.19	12120	-449	19.97
0.3	400	3	394	3	11800	-440	19.7
0.3	400	40	394	39.7	11925	-445	19.8

Table V-1. Effects of V_{qcom} and V_{dcom} Changes on an Isolated Inverter's Load Voltages and Real and Reactive Power Output.

When the same test is conducted with two inverters paralleled, much different results are obtained. Initially two inverters are commanded to the same values of q and d voltage as seen in the first data row of Table V-2. The data in this table is only shown for one of the two inverters to simplify the comparison with Table V-1. Subsequently, the V_q voltage of the first inverter increased by 5 V while the second inverter's commanded values are not changed. The bus voltage only increased by 2 V, but the real power output of the inverter almost doubles. V_q is then restored to 400 V on the first inverter and V_d is commanded to 3 V on the first inverter. The real power output of the first inverter remains the same, but the reactive power output increases from –215 to 2718 kVARS.

R_2	V_{qcom}	V_{dcom}	V_{Lq}	V_{Ld}	R_{pwr}	Q_{pwr}	I_2
0.3	400	0	397	0.07	5946	-221	10
0.3	405	0	399	0.06	11131	-215	18.3
0.3	400	3	397	1.53	5946	2718	11

Table V-2. Effects of V_{qcom} and V_{dcom} Changes on One Inverter's Load Voltages and Real and Reactive Power Output when Paralleled With A Second Unit.

It is important to understand that changes in the first inverter's power output cause opposite changes in the second inverter's power output even though the second inverter has not changed its q or d voltage. The load has not changed and the bus voltage is only slightly varied, therefore the total power required remains relatively constant. Initially each inverter was delivering approximately 6 kW of real power to the load for a total of 12 kW. When the q-voltage of unit one is raised causing its power output to increase to 11 kW, the power output of unit two decreased to 1 kW. The same effect is true of the reactive power loading.

This data supports earlier claims for voltage control as a means to share power. Varying the q-voltages changes the real power sharing while varying the d-voltages changes the reactive power sharing. It also shows that an inverter control could easily assess whether other inverters are online by simply performing a quick voltage-power sensitivity analysis. If small changes in the q or d voltage do not significantly change the inverter's power loading, it must be alone on the bus. One more piece of local information is now available to the unit.

Experiments conducted demonstrated that if the inverters all have the same line resistance, sharing between them is automatic. The interesting and more realistic case is when paralleled inverters have different connection line impedances. Two inverters were paralleled with one inverter at the expected nominal line impedance of 0.07 Ω (R1) and the other at half this value or 0.035 Ω (R2). No sharing control was implemented. The inverter controllers simply maintained their nominal filter capacitor voltage. The bus load initially was a 20 Ω load at a 0.8 PF. At time .02 the load changed to 10 Ω at 0.8 PF. If the inverters shared the load exactly they would stabilize at just below their 15 kW rated capacity. As seen in Figure 15, the inverters fail to share the load and the lower impedance inverter overloads. A sharing mechanism must be implemented when the line impedances are not matched and a load approaches the total capacity of the interconnected inverters.

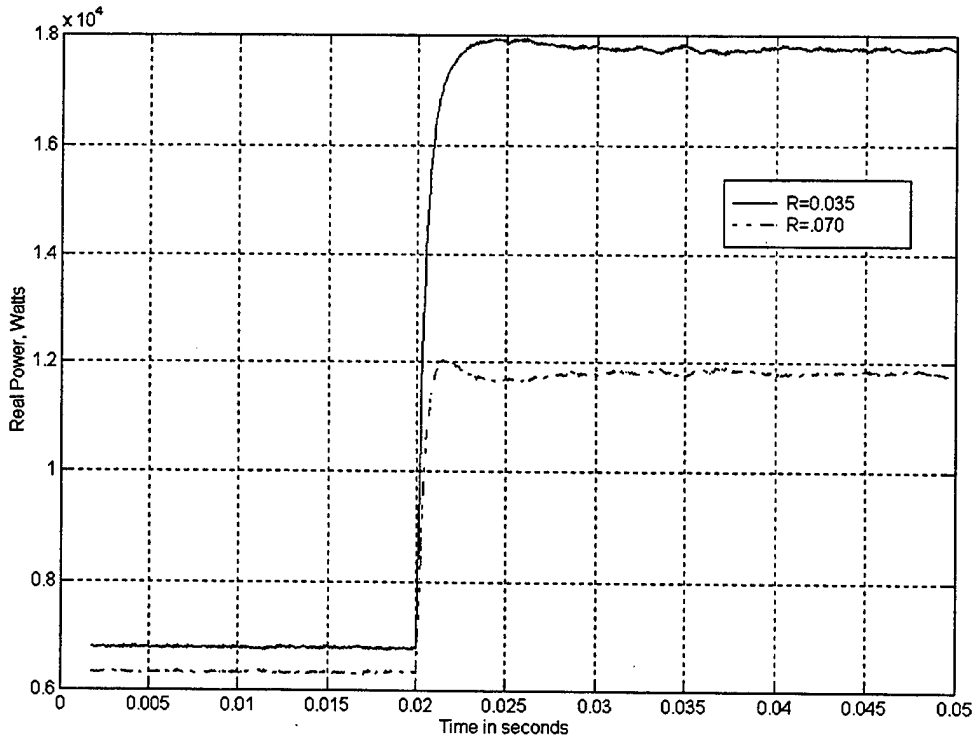


Figure 15. Real Power Sharing between Two Inverters with Line Impedances of 0.035 and 0.07 Ω for a Step Change in Loading with No Sharing Control.

B. AUTONOMOUS VOLTAGE DROOP

The autonomous method tested for paralleling inverters uses voltage droop as the real and reactive power sharing method. This is in sharp contrast to the typical control of generators using frequency droop to control the real power sharing and voltage droop to control the reactive power sharing.

1. Frequency Droop

Many journal articles are available covering techniques to parallel inverters using frequency-droop to control real power sharing [2],[4]-[6],[12]. After the frequency of the inverters has been reduced to accommodate sharing, the no-load setting for the frequency must be raised to restore the system frequency. Restoration of the system frequency can

be difficult in a distributed system. Frequency droop techniques were tested using a ±1 Hz range to represent zero to one hundred percent loading. As the inverter's load increased the PWM sinusoidal control frequency was reduced thereby reducing the inverter's frequency output. In this thesis voltage-droop control is explored as an alternative to frequency-droop control.

It is not as apparent to the casual observer how frequency droop works. How does raising the no-load frequency of one inverter cause that inverter to increase the frequency of the entire bus and take on more power? A voltage is produced in the stator winding of an AC generator by spinning the rotor and impressing a DC current into the rotor windings. The resulting current that flows in the stator windings establishes a rotating magnetic field that rotates in unison with the rotor. The angle between the rotor and stator magnetic fields is called the power angle and is directly related to the amount of real power produced by the machine. In a system of generators supplying a common bus, the operation of a single machine is analogous to a small and large gear. The inertia of the large gear tends to keep the small gear locked in step. That is, when a generator tries to increase its frequency, it "pushes" against the magnetic field of the system increasing the angular separation between the rotor and the bus voltage. If the other generators connected to the bus are of approximately the same power rating or less, the system frequency will rise in response to this "pushing". The other generators on the system are also locked onto the system magnetic field so they will speed up without any change in their applied rotor torque. This speedup for a constant prime-mover torque means that the angle between the torque and field vector on their rotors is reduced, therefore the power output is reduced. Finally, we have the entire system operating at a higher frequency with one unit supplying more of the power.

For a system of interconnected inverters this locking-field analogy is not applicable. It is a solid-state machine. A switch knows nothing about magnetic fields, torques, and motoring. It would seem that the frequency droop technique would fail for these solid-state inverters, but it does not as is shown in [2],[4]-[6],[12]. As the power output of the inverter increases, the frequency droop causes a reduction in the phase angle

of the voltage relative to the system. As the inverter's phase-angle decreases, its current output decreases, and its power output decreases.

2. Voltage Droop Justification and Operation

When using frequency droop one disadvantage is that the system frequency must be restored, whereas with voltage droop the frequency is maintained at 60 Hz. Many loads are far more sensitive to frequency shifts than voltage shifts, so having a predictable bus frequency is advantageous. Another reason for trying a new droop technique is the sensitivity of the frequency droop technique to noise. Considerable noise was generated in the frequency droop control when using the power calculated at the inverter's filter-capacitor in the droop equation. This noise generated oscillations that slowly increased over time. Filtering the required power feedback signals could reduce these oscillations but would simultaneously reduce the controller's response time.

Using the inverter design from Chapter III, two inverters were paralleled using frequency-droop control. The frequency-droop control was enhanced with automatic frequency-recovery control. The design of this frequency controller was modeled after techniques described in [4]. The ACSL code and details of the simulation can be found in Appendix B. These paralleled inverters were subjected to a step increase of 43% to 100% loading. In Figure 16 the effect of this step-increase in load in shown on Inverter #1. The inverters respond very fast with little overshoot, but oscillation caused by noise in the filter can be as large as 20% of the inverter loading in steady state. The oscillation's size becomes relatively constant after the automatic frequency restoration is complete.

One reason for this noise sensitivity is probably due to the importance of the individual frequencies in the synchronous reference frame transformations. Errors and noise in the drooping frequency values are multiplied in the controller and the system model equations.

Therefore, with voltage-droop control, frequency restoration is eliminated and frequency noise feedback is avoided while providing a method to share the real and reactive power between inverters.

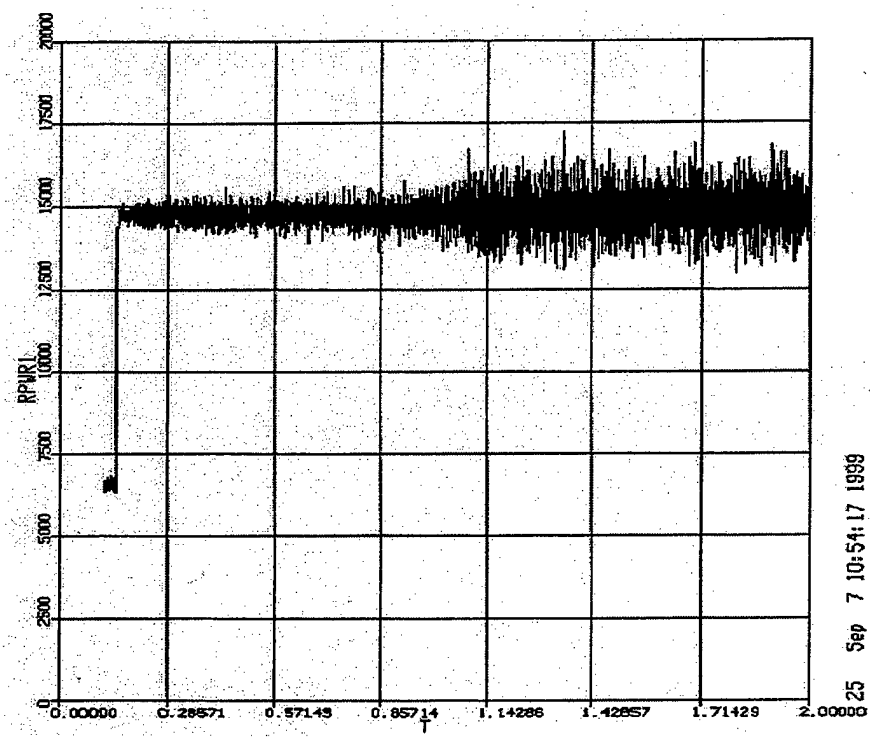


Figure 16. Noise and Oscillation in the Frequency Droop Power Sharing Technique.

Voltage droop control in the synchronous reference frame is essentially a nonlinear DC circuit analysis problem as shown in Figure 17. The figure shows three parallel inverters in the synchronous reference frame. As shown earlier in this thesis, the synchronous reference frame manipulates the peak values of voltage and current. These peak values can be thought of as DC quantities.

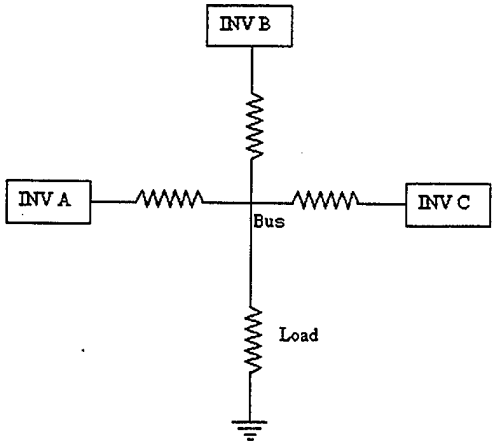


Figure 17. A Synchronous Reference Frame DC Representation of Three Paralleled Inverters with Balanced Loading.

Each of the inverters is regulating its output voltage. When the inverter is supplying power to the load, a current flows from the inverter through a line resistance to a common bus. If the line resistances are relatively small or the currents through them are small, then the inverter output voltage and the bus voltage are only slightly different. Thus, the bus or load voltage is controlled.

The load voltage is generated by the sum of the currents from the inverters passing through the load resistance. An inverter supplies current if its output voltage is greater than the load voltage. Current will be supplied to an inverter if its voltage is less than the load voltage, thus making the inverter a load. By varying the inverter output voltage relative to the load voltage, the amount of current and power from that inverter can be controlled. One problem with this logic is that only local parameters are available to the inverter so the load voltage is unknown. Another problem is that voltage is already being controlled to maintain the load voltage. The question this brings up is can the inverter output perform voltage control and real power control simultaneously? The answer is yes under certain conditions.

An inverter can calculate its local power output. The inverter controller can be programmed to reduce its output voltage as its power output increases. If no other

inverters are operational, this local voltage drop will only cause a corresponding bus voltage drop. If other inverters are present on the bus, this voltage drop will cause a much less significant drop in the load voltage. The load voltage drop is limited since it is dependent on the currents from all the inverters. Thus, the ΔV across the line resistance of the drooping inverter decreases causing it to supply less current and power, while the ΔV across the line resistances of the other inverters has increased causing them to supply more current and power.

The voltage droop sharing relies on the system having a voltage range in which it can operate. As discussed previously, the allowable range for AC bus voltages for a naval ship application is approximately ±10%. This range will prove to be more than adequate.

A problem voltage droop sharing has is that it is linked to the line resistance which is an unknown. Inverters with smaller line resistances are harder to control than inverters with larger line resistances. A one volt change in voltage across a small output resistance causes a larger change in current output than a one volt change across a larger resistance. From [14] using a minimum of 14-gauge wire, it was found that the resistance is approximately 2.575 Ω per 1000 feet, and for 12-gauge wire the resistance was 1.619 Ω per 1000 feet. Assuming output cabling runs of approximately thirty feet and an average resistance of 2.1 Ω per 1000 feet yields a line resistance of 0.07 Ω. This line resistance can be used as a starting point for sharing analysis.

3. Controller Modifications

The original controller ensured that the inverter output voltage was maintained using a two-loop controller. A new loop will be added to this controller for the voltage-droop sharing feedback. For the controller, the entire 90 V range allowed for the 450 V bus is not required. The droop range of voltages must be large enough to allow the controller to compensate for variations in connection resistances between the units and the load bus. If the range is selected to be too large then control may be slow and large

load transients may produce unacceptable power sharing transients as will be shown later. A droop-control range of about ±1% (450 ± 5 volts) is used assuming a 15 kW inverter. A desired commanded voltage will be computed as in Equation (33).

$$Desired\_q = 455 - \frac{Pwr \cdot 10.0}{15000.0} \tag{33}$$

This Desired_q is then compared to the current value of the commanded q-voltage. If they differ by more than 0.5 volts and Desired_q is greater than the currently commanded voltage, then the commanded voltage is increased. If the desired voltage is less than the commanded voltage, the commanded voltage is reduced. The algorithm for determining how these changes are implemented is discussed in the testing section of this chapter.

C. TESTING THE AUTONOMOUS DESIGN

Testing the autonomous design was done in steps. Initially, the inverters were paralleled without any sharing control. The sharing control was then actuated with various step sizes and commanded voltage algorithms. The next step was to test the stability, response time, and sharing ability of the inverters. Finally, inverter start-up and synchronization was explored.

1. Step Size Determination

For this section only real power control is addressed. The commanded q-voltage controls the real power sharing between inverters as discussed previously. Stepping the commanded q-voltage is a key to the proper sharing of the real power. This is not a controlled restoration as was discussed in [5], since a parameter such as frequency is not being recovered to a previous value. The reason for stepping the voltage changes is that the inverters are not in communication with each other. The dynamics of each correction effects the final desired value of the commanded q-voltage. For example, as the inverter reduces its voltage its power output decreases, thus the Desired_q value changes. Taking

large steps will cause uncontrolled oscillations and overshoot. Testing demonstrated that large growing oscillations were much more likely to occur if inverters are commanded to a voltage less than the load voltage. A reverse-powered inverter acts as a very large reactive load to the system. Its low impedance can easily cause the other inverters to be overloaded in milliseconds.

The voltage droop effects the Desired_q value, but also the corrections being performed by other inverters on the bus also will effect the local Desired_q. Therefore, big corrective steps must be avoided since there is no direct knowledge of other units on the bus. But, if the steps are too small then the corrective action may take too long. Typical values for overload times for similar three-phase inverters such as the Toshiba 4100 series 25 kVA are 125% for 90 seconds and 150% for 30 seconds [15]. These same overload specifications will be used for evaluating inverter performance.

Due to the conflicting requirements of a large step size for speed and a small step size for stability, many variations were tried. The first attempt used a linear step size. Generating a suitable step for large line resistance variations with a linear step size proved difficult. A variable step size was tried next. Using the difference between the existing commanded filter voltage and the desired droop voltage gives a variable step size. This variable step size can be scaled by a conditioning factor to further control the step. Thus, inverters whose output voltage is further from the desired droop voltage adjust more than inverters whose output voltages are closer to the desired droop voltage. Therefore, the convergence to the desired droop value is exponential. Experimentally a step-size conditioning factor of 0.07 was found to give good results. The equation to calculate the new commanded voltage is given in Equation (34) where the Desired_q value is the result of the droop calculation.

$$V_{Qcom,new} = V_{Qcom,old} + 0.07 \cdot \left(Desired\_q - V_{Qcom,old}\right) \tag{34}$$

This one equation works for both step-up power and step-down power transients, since the difference of the quantity in parentheses sets the sign of the factor added to V_{Qcom}.

Tests presented earlier in Chapter III on the individual inverter showed that the transients caused by changes in V_{Qcom} lasted approximately 10 ms. To prevent oscillations, a new V_{Qcom} is only calculated every 10 ms. This delay insures that transients caused locally by the changes in the commanded q-voltage are not misinterpreted in the inverter's sharing loop. The negative aspect is that this sets a ceiling on the sharing response time of the controller at 10 ms.

The worst case scenario for control is when all the inverters step to the next value of V_{Qcom} at the same time since they have no time to sense each others changes on the network. In the simulations this simultaneous-step scenario will always be used to test worst case response.

2. Load and Line Variation Testing

In Figure 15, at the beginning of this chapter, two inverters with different line impedances were paralleled then the common-bus load was step changed to a lower impedance. That same simulation is run again but this time the voltage-droop feedback-sharing loop is added to each of the unit's controllers. An instantaneous load demand change from 43% to 100% was used to test the autonomous voltage-droop. A load change of greater than 50% takes the system from a state where one inverter could handle all the load to both inverters being required to operate at full-rated power. A 57% change was arbitrarily chosen. The results of this power demand increase are seen in Figure 18. The initial behavior of the units from 0.02 seconds to 0.03 seconds is similar to what it was without the droop control. At 0.03 seconds the first calculated step change in the inverter q-voltages is applied. The exponentially stepped droop causes the inverters to converge towards an average value. Neither inverter has knowledge of this average value so the convergence is not complete. The powers for the units steady out at 14 kW and 15 kW for a difference of about 7.5%. The load voltage for this example stabilized at 447 volts. Consequently, for paralleled inverters that have one unit with double the line resistance of the other unit, sharing is achievable within 8%. The peak power achieved is

about 18 kW (120%) for 10 ms, which is much less than the minute allowed in this overloaded condition. At these peak powers the droop control could be augmented with an overpower correction which will droop an inverter's voltage prior to the inverter shutting down due to thermal overload. This concept is covered later.

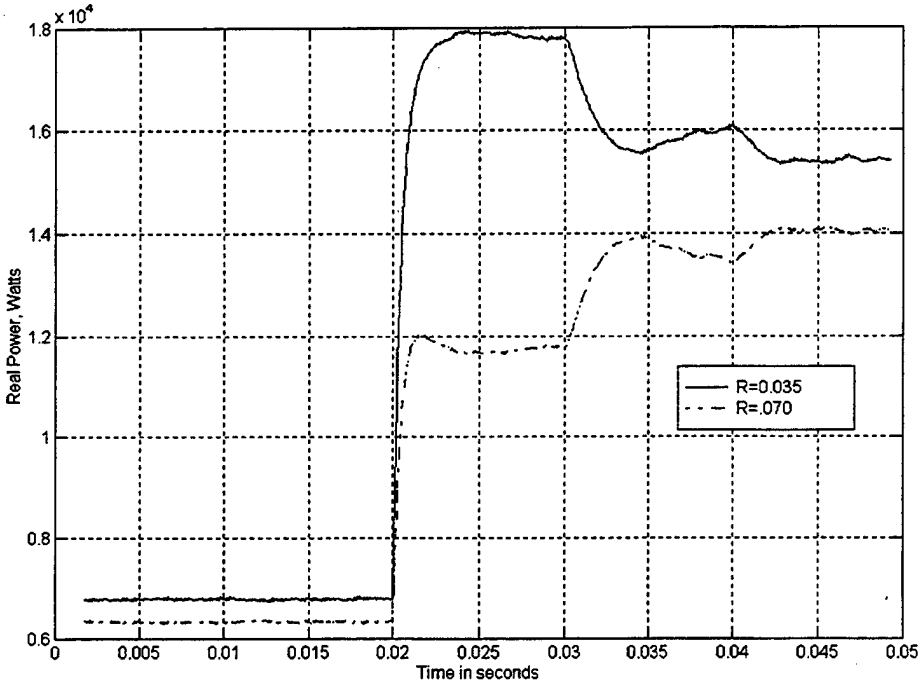


Figure 18. Real Power Sharing Between Two Inverters with Line Impedances of 0.035 and 0.07 Ω for a 43% to 100% Step Change in Loading.

The next experiment tested the sharing loop's ability to handle a step change in line resistance for inverter #2 from .035 Ω to .007 Ω (1/10 of R_1). This event could be analogous to a large load shifting such that it is now physically closer to inverter #2 on the bus. The resistance of inverter #2 was changed at time 0.02 as shown in Figure 19. One-tenth of a second later the first sharing loop compensation is applied. This first compensation brings the units very close to their original sharing status. After four corrective steps the units stabilize with a 12% power difference. This test shows that the sharing accuracy is linked to the line impedance difference.

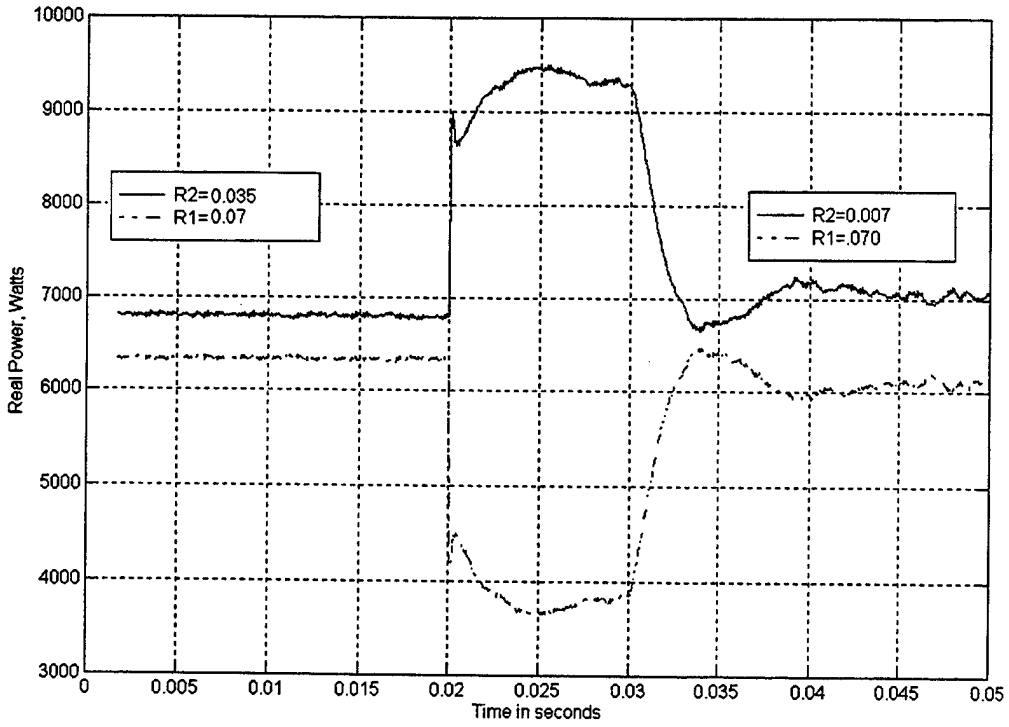


Figure 19. Effects of Shifting the Line Impedance of Inverter Two from 0.035 Ω to 0.007 Ω.

The line impedance effect on sharing varying loads was now tested with R_1=0.07 Ω and R_2=0.007 Ω. The load is stepped changed from 43% to 100% at time 0.05 seconds as shown in Figure 20. The first noticeable difference from the case where the resistance differed by half is that the overshoot for the lower impedance unit is greater. In the 0.035 Ω case the overshoot peaked at 18 kW, but now it is peaking at just over 22 kW. This is the 150% overload point for the inverter but once again the inverter stays at this power for a mere 10 ms which is far shorter than the allowed 30 seconds by more than three orders of magnitude. The first sharing step causes this inverter to drop just below its full rated power of 15 kW. Subsequent corrections cause it to rise above its full rated load. Steady operation is achieved in 25 msec. At this stable operating value, the inverters were sharing the real power within 12% with a bus voltage of 448 V.

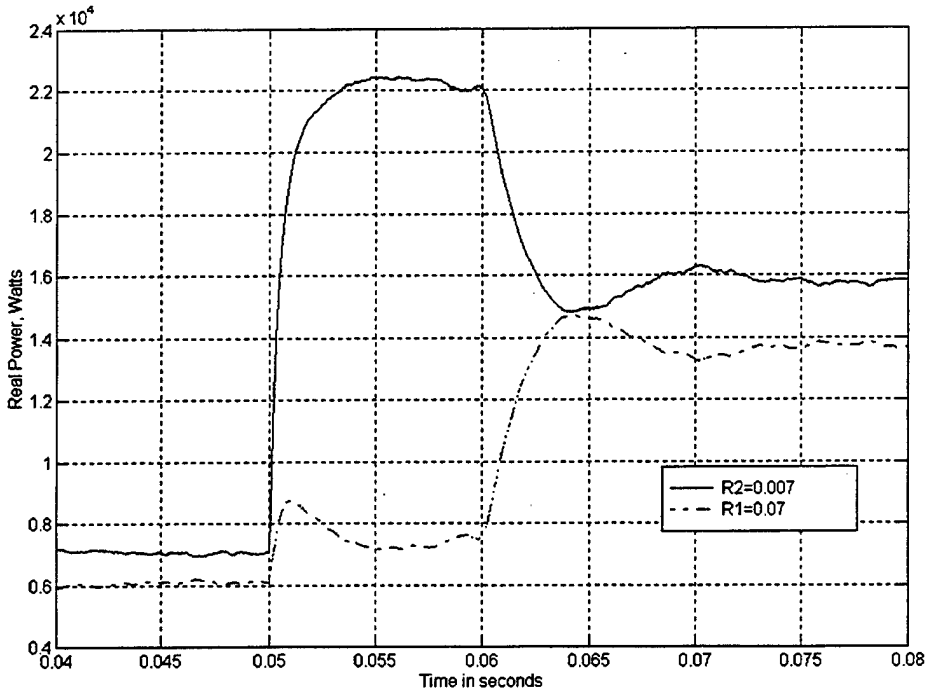


Figure 20. Real Power Sharing Between Two Inverters with Line Impedances of 0.007 and 0.07 Ω for a 43% to 100% Step Change in Loading.

With the inverters at this high power, the load impedance was restored to a value that should load each inverter at just above 43% of their rated load (6.5 kW). The load step was programmed at time 0.1 seconds and the results are shown in Figure 21. Again the lower impedance unit with R_2=0.007 Ω overshoots the desired value of power but maintains a positive power flow. The main transient effectively stabilizes after 25 msec but the inverters continue to wander slightly. The inverters stabilize with a power difference of only 600 W (10% difference) at a bus voltage of 449.2 V.

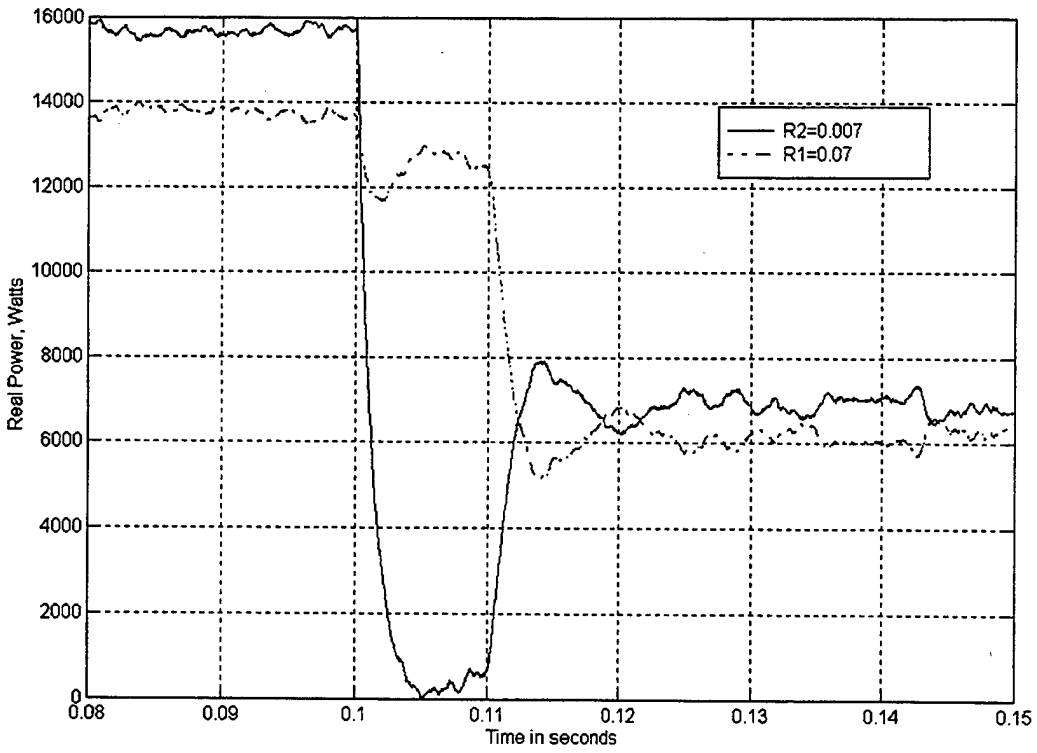


Figure 21. Real Power Sharing Between Two Inverters with Line Impedances of 0.007 and 0.07 Ω for a 100% to 43% Step Change in Loading.

The difference in resistances was now widened to two orders of magnitude with R1=0.07 Ω and R2=0.0007 Ω. This is an extreme case that would only occur if one unit was bolted to the load and the second unit was located more than 30 ft away. The results of a 43% to 100% step increase in power for these impedances are shown in Figure 22. Once more the unit with the lower line resistance (R2=0.0007 Ω) overshoots during the step increase in load power. This time it exceeds the 150% overload limit of 22 kW and reaches 24 kW. With resistance magnitudes being so low and having two orders of magnitude difference between them, damped oscillations begin. In previous examples the transients steadied out after approximately 25 msec, but with the oscillations the time to steady state increases to 55 msec. The power sharing stabilizes at a reasonable 13% difference with a bus voltage of 445 volts.

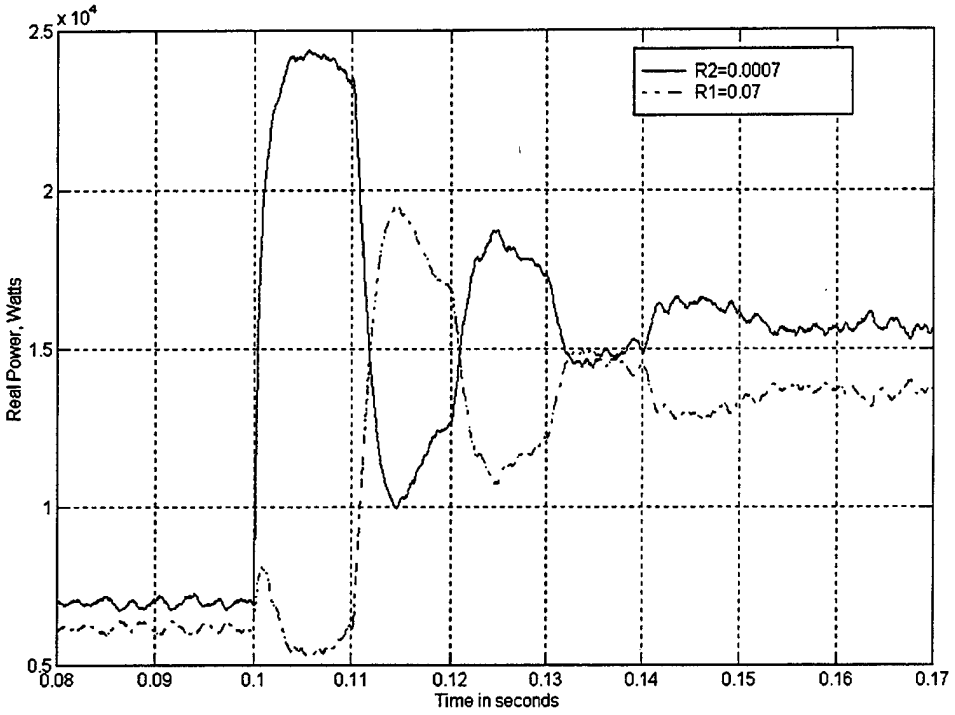


Figure 22. Real Power Sharing Between Two Inverters with Line Impedances of 0.0007 and 0.07 Ω for a 43% to 100% Step Change in Loading.

The step decrease in power from 100% to 43% was also tested for these resistances. Again the large oscillation is present for the inverter with the smaller resistance. This time the overshoot is enough to send inverter #2's power negative but only for one cycle of the oscillation. Even with the undesirable oscillations, the system stabilizes in 55 msec with a 14% variation in power sharing and a bus voltage of 448.5 V.

Considering that an impedance difference of two orders of magnitude should never occur and that the units still shared the load within 14%, this test was considered very successful.

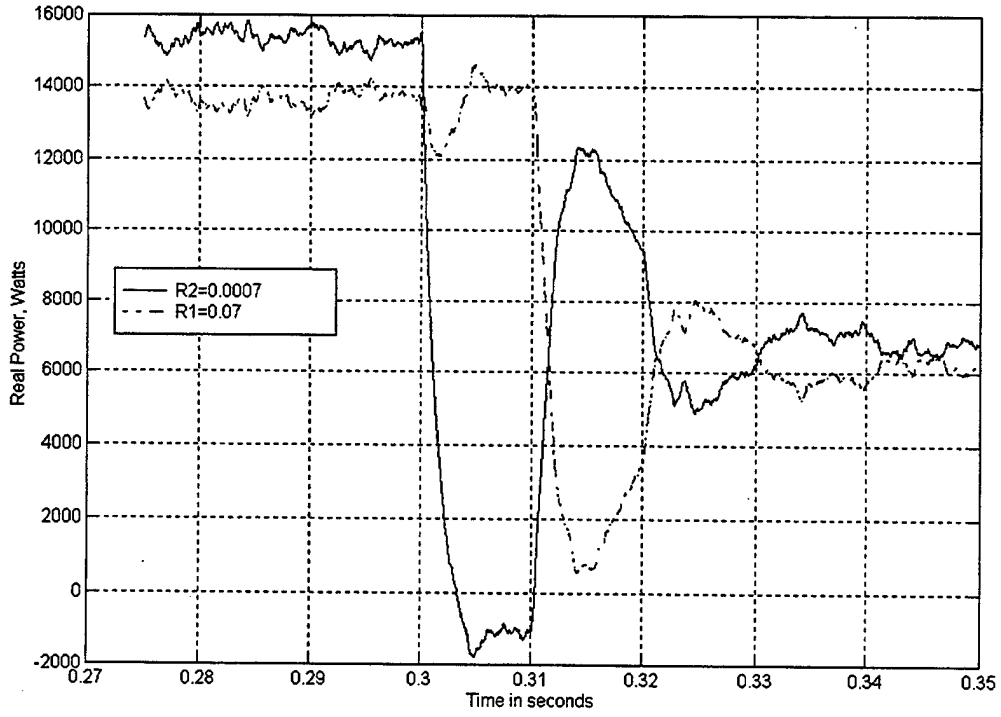


Figure 23. Real Power Sharing Between Two Inverters with Line Impedances of 0.0007 and 0.07 Ω for a 100% to 43% Step Change in Loading.

Oscillations are generally bad especially at or above the power rating of the unit. But compared to the non-oscillating cases, there is an advantage. The oscillation about the unknown mean tends to cause the final sharing discrepancy to be less pronounced as can be seen in Figure 23.

3. Further Modifications and Observations

The worst case scenario of large impedance differences, 50% step changes in power, and simultaneous voltage corrections may be too unrealistic for these fast-acting inverters. But, additional modifications to the controller could be used to minimize their impact. The first is termed 'Early Step'. By incorporating Early Step, overload time could be minimized for jumps in power above 150%. If the power of an inverter ever exceeds the 120% mark, then its first corrective voltage droop step is taken within 2 msec

of the transient rather than after 10 msec as it does normally. Not only does this lower the power on the overloaded unit faster, but it will give the other unit time to recalculate its droop step in response to load changes on the bus (see Figure 24). This simulation study shows a significant improvement over the original run of Figure 22. The first big improvement is the reduction of the magnitude of the overshoot from greater than 24 kW to 22.6 kW or just over 150% load.

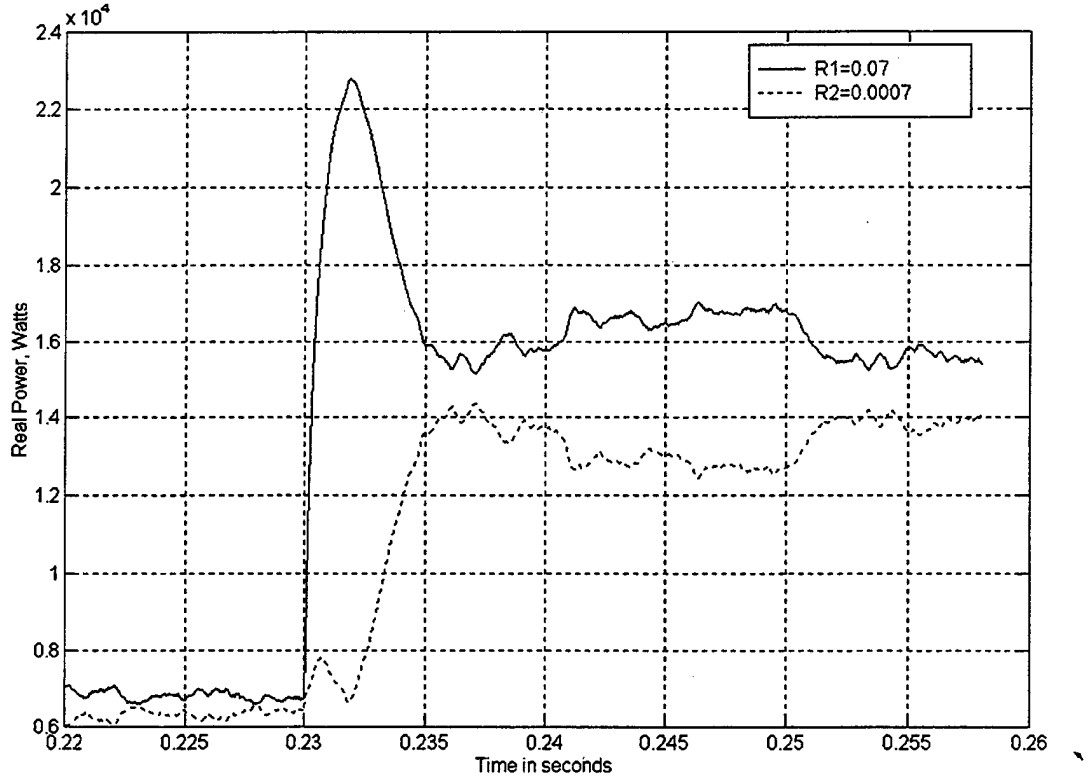


Figure 24. The Simulation of Figure 22 Rerun with the Early Step Modification.

The next improvement is the time spent above the 15 kW power rating of the inverter. Five milliseconds after the peak overshoot, the inverter is at its 15 kW power rating. This is twice as fast as before. Another very noticeable change is the absence of large oscillations. The loss of oscillations is attributed to Early Step causing the inverters to shift commanded voltages out of synch with each other. Therefore, Inverter #1 has feedback that it can use to calculate a better next step. Twenty-five milliseconds after the transient the inverters stabilize with a 9% power deviation at a load voltage of 447 V.

The stabilization time is twice as fast. These numbers are now very similar to the case where one inverter's line impedance was one-half the other inverter's line impedance. In order for this technique to work the controller must respond within two milliseconds which should not be a problem.

To minimize long term overloading at values just above the inverter's rated capacity another controller modification can be made. The droop control would be modified to continue to droop an inverter's voltage if the inverter is more than 100% loaded. This additional droop would continue until the inverter loading was less than 100%, or until the local voltage was drooped to 440 V. The 440V limit is five volts lower than the full load expected limit of 445 V. If no voltage floor was given and a single, overloaded inverter was on the bus, this inverter would droop the voltage of the bus until loads started failing due to undervoltage.

As stated previously, only real power sharing was tested. For naval vessels this is satisfactory since they ensure major loads have power factors greater than 0.8 lagging. For commercial applications the controller could be modified to share the reactive power also. The same droop technique could be used on the commanded d-voltages to share the reactive power.

A final observation is the fact that resistances in power systems tend to demonstrate some non-linearity, and the DC supply is not an infinite source. The resistance feedback present in all power circuits is not accounted for in these models. The non-linear resistance characteristics of power transmission systems will be more pronounced for very low impedance connections since these connections would experience the largest surges in current flow. Although a second-order effect in actual testing, it may allow the controller to work even better. Additionally, the DC source is modeled as a hard source when in reality it is not. Thus, the low-impedance inverter will cause a larger drop in its DC source voltage since it will demand a larger current. The inverter's controller is forced to compensate for this voltage drop thereby slowing its reaction time. The overall effect will be a reduction in the peak power value attained during the excursion.

4. Starting a Unit

The inverters were tested considering step-up power transients and step-down power transients in the previous section. Next, the ability to parallel an operational inverter with a newly started inverter was investigated.

Two options for paralleling a previously idle inverter with an operational inverter were investigated. The first option was to assume that the unit 's filter was always online and the breakers were located immediately after the switches. The filter would therefore be set up for bringing the inverter online. The sensors would all have initial values to work with so the controller would be ready to operate as soon as the breakers were shut. Additionally, the frequency information would be available from the filter capacitor. As an experiment, R_1 and R_2 were set at 0.03 Ω and the effect of isolating the inverter #2 at the switches was tested. In the simulation this isolation is achieved by forcing the filter inductor current to remain at zero while the "breakers" are open. The results were disastrous. The filter became a 125 kVAR load on the bus, and bus voltage dropped to 225 V. The simulation ran longer than a real inverter would have since the operational inverter's output current was 600% of its rated current. Another problem with this approach was that it actually put the controller in a worse situation than if the inverter was started with no values available. The reason was that the power flows were initially in the opposite direction since a filter dangling on the bus behaved like a highly reactive load. Thus, the control system had to compensate excessively for the errors introduced. These complications lead to highly oscillatory and unstable characteristics. Although a failure, this experiment does show a need for protection from this casualty. If one inverter suddenly drops offline at the switches, its filter must be taken offline. Thus, the breakers for the system must isolate the inverter and its filter.

The second choice was to have the breakers simulated as being after the filter on the line. This positioning would allow the inverter to be started up in an isolated manner

shielded from the bus by the breakers. The frequency of the oncoming unit would have to be matched to the system frequency using phase-lock loops located on the output side of the breakers or manually. Placing the simulated breakers on the inverter's filter outlet yielded much better results. As long as the AC paralleling conditions are met before shutting the breaker, the inverters experience a transient then steadied out at their shared power levels.

D. PARALLELING MORE THAN TWO INVERTERS

The sharing algorithm was tested with multiple inverters to verify its robustness and scalability. In this section the modification to the inverter model is discussed, and some simulation results are presented.

1. Model and Controller Modifications

The model and controller required no modifications to work with more than two inverters. The extra inverters were added to the simulation without the need to change any system setting or model constants. Up to six inverters have been paralleled using the voltage droop technique, but to keep the graphs legible the results for three parallel inverters are described. Details of this model can be found in Appendix A.

2. Results

The parameters used for the simulations are the same as those reported in the study cited in Figure 18 except that the third inverter is added with a line resistance of 0.05 Ω. The same load shift used to produce Figure 18 was used with three inverters. This caused the inverter loading to shift from 28% to 66% of their rated capacity. The results in Figure 25 look very similar to the two-inverter case. Once again the lower line impedance units try to take a larger share of the power, but the droop control forces them

to share the load. The inverters stabilize in about 25 ms, sharing the load within 8% at 449V.

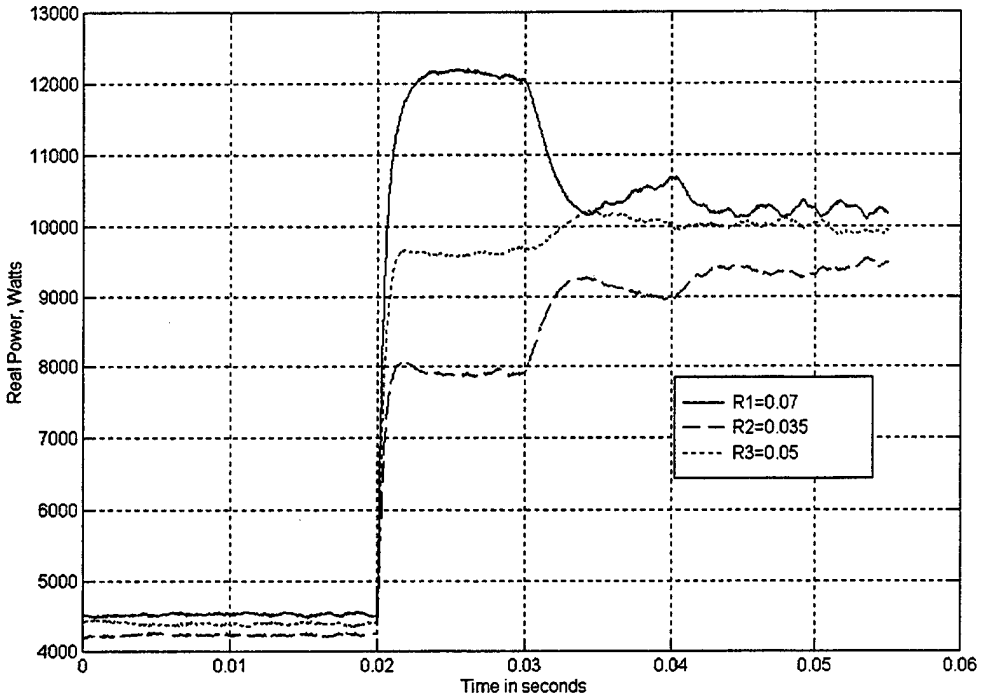


Figure 25. Real Power Sharing between Three Inverters with Line Impedances of 0.035, 0.05, and 0.07 Ω for a 28% to 66% Step Change in Loading.

For a step decrease in load the same trend is seen. The lower resistance units tend to overshoot the sharing value, but the droop control brings them back together. For the step decrease in load shown in Figure 26 the sharing once again is excellent. The transient stabilizes in approximately 25 ms with the units sharing within 5% of each other at 450 V.

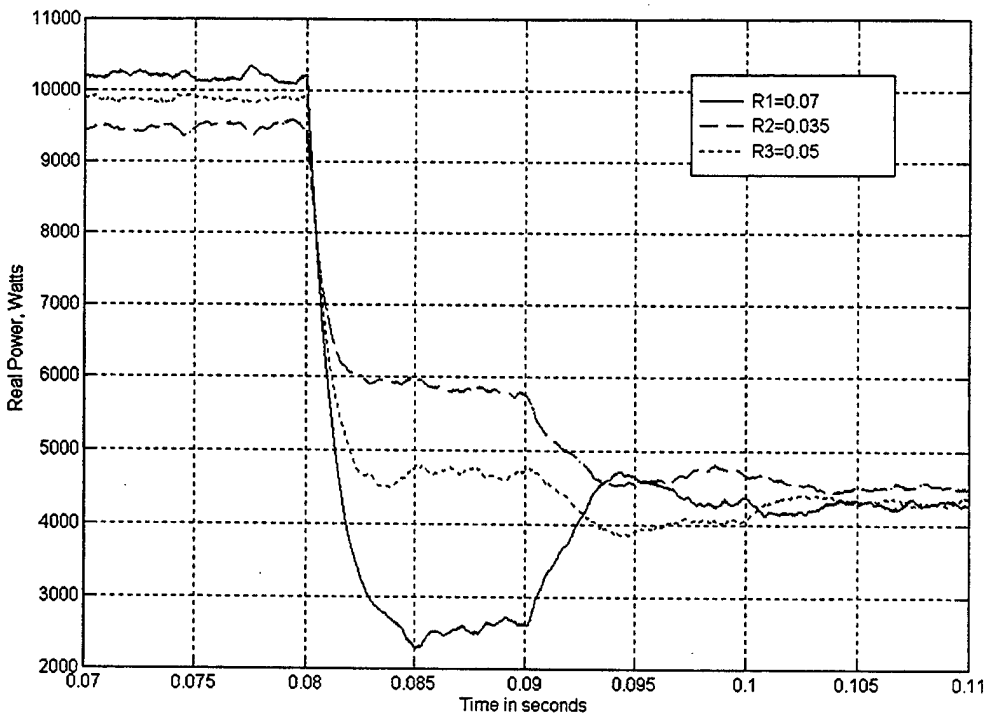


Figure 26. Real Power Sharing between Three Inverters with Line Impedances of 0.035, 0.05, and 0.07 Ω for a 66% to 28% Step Change in Loading.

VI. RECOMMENDATIONS AND CONCLUSIONS

This final chapter contains a description of appropriate extensions to the presented thesis work, other areas of related work for parallel inverter control, and a general conclusion.

A. CONTINUED WORK

Due to time and material limitations, only a single inverter unit was ever built in hardware. The next logical step is to test the results of this thesis using actual power converters and DSP control resources such at the dSPACE control card. If this proves cost prohibitive, a detailed Spice model would also prove valuable. Variables such as sampling rates, sampling accuracy and DSP inaccuracies must wait for a hardware model to ensure accurate assessments.

A Spice model may be useful in determining the source of the large filter currents simulated, and discover if they truly exist and if their minimization has an adverse effect on the inverter controller operation.

B. OTHER STUDY

The signal-injection techniques described in this thesis may prove useful as a refinement to or backup for the autonomous droop control. In simulation testing an injected average power signal was detectable by the individual inverters and allowed sharing to within three percent. This signal could be used to periodically augment the autonomous system in order to bring the sharing percent to below three percent for all steady-state conditions. Additionally, signal injection could be used as a backup if droop is not desirable and a very tight voltage range is required.

Other power line communication techniques such as CEBus could be further investigated not only as an inverter sharing mechanism, but also as a means to communicate large amounts of system data from any machine powered from the ship's

power grid. A power line intranet could be used to coordinate sharing and efficiency optimization between components in the engineroom. System data would also be more available to review. For example, an engineer in a forward space could simply plug in his modified laptop and be shown on a GUI all the power equipment running in the engineroom and the equipment's status.

C. CONCLUSION

In this thesis a distributed control method for paralleled inverters was developed. A robust and fast single inverter model, filter, and controller was designed. Many sharing techniques were examined for use in paralleling inverters in a distributed network. An autonomous voltage-droop method augmented with early stepping was developed. This voltage-droop method was simple to implement, nominally required 25 ms to stabilize, and allowed the real power to be shared between paralleled inverters within 8% of each other while maintaining a constant system frequency.

APPENDIX A: ACSL AUTONOMOUS DROOP CODE

```
PROGRAM   ! Share3.CSL
! Parallel three inverters with voltage droop control only

INITIAL
      MAXTERVAL maxt = 1.0e-6          !"max integration step size norm=e-5"
      MINTERVAL mint = 1.0e-8  !"min time step for int algorithm norm=e-7
      CINTERVAL cint = 5.0e-5          !"data communication interval"
      ALGORITHM ialg = 5               !"4rth order runga kutta"
      NSTEPS nstp = 1                  !" 1 = int step size controlled by maxt"
      CONSTANT tstop = 0.1             !"stop point for integration"
      PARAMETER (twopi = 6.283185307)       !" 2 times pi = omega times period"
      CONSTANT rad120 = 2.094395       !" 120 degrees in radians"
      CONSTANT we1 =377.0              ! 60 hz system
      Constant we2 =377.0
      Constant we3 =377.0
      !"DC Source Constants"
      CONSTANT vi = 950.0              !" first inverter DC input"

!"Switch Description, only describe top switches, bottom complimentary, 3 phase"

      LOGICAL SA1,SB1,SC1              !"Inverter1/2 switch pairs"
      SA1 = .FALSE.                    !"Top switch open, bottom switch shut"
      SB1 = .FALSE.
      SC1 = .FALSE.
      LOGICAL SA2,SB2,SC2
      SA2 = .FALSE.                    !"Top switch open, bottom switch shut"
      SB2 = .FALSE.
      SC2 = .FALSE.
      LOGICAL SA3,SB3,SC3
      SA3 = .FALSE.                    !"Top switch open, bottom switch shut"
      SB3 = .FALSE.
      SC3 = .FALSE.

      !"Carrier Triangle Constants, symetric, bipolar triangle"
CONSTANT ftriang = 10800.0       !"triangle freq in Hz, want this > 9*fsin" (10K)
      Vtripk = vi/2.0            !"peak triangle amplitude"
      Ttri = 1.0/ftriang         !"triangle period in sec"
      Tover2 = Ttri/2.0          !"triangle half period"
      slope = 4.0*Vtripk*ftriang !"slope of the triangle,1/4 of period"

      CONSTANT LF=5.626977e-05       !"inverter one output inductor(.004)
```

```
        CONSTANT CF=1.1253954e-02
              !"bank capacitor value (e-4)
        CONSTANT RF=.1                  ! filter resistance
!!
        CONSTANT R=.07
        CONSTANT L2=1.0e-6              ! series line inductance
        Constant R2=.035                ! inv2
        Constant L22=1.0e-6
        Constant R3=.05                 ! inv3
        Constant L23=1.0e-6

        !"Phase load constants"
        CONSTANT RL=20.0                !"resistor load in each leg"
        CONSTANT CL=1.0e-5              !"load capacitor value (e-6)
        CONSTANT LL=.021                !"0.021 inductor in each load"

        Constant onoff22=1.0        ! 0.0= #2 inverter off line
        Constant onoff23=1.0        ! 0.0= #3 inverter off line

        ! sharing values
        Constant Vqcom1=450.0           ! No-Load commanded voltage
        Constant Vdcom1=0.0
        Constant Vqcom2=450.0
        Constant Vdcom2=0.0
        Constant Vqcom3=450.0
        Constant Vdcom3=0.0

        Constant sharetimeR=.05         ! (.05)1st Update for real power sharing loop
        Constant Vstep=.07          ! voltage stepping size (.5=critical damp)
                                    ! with lower r2 need lower
        Constant onoff1=1.0         ! 0.0= #1 sharing off.
        Constant onoff2=1.0         ! 0.0= #2 sharing off.
        Constant onoff3=1.0         ! 0.0= #3 sharing off.

END   !"initial"

DYNAMIC
        !"This section is executed each cint seconds"
        !"This statement stops simulation at tstop seconds"
        TERMT (t .GE. (tstop-0.5*cint))

        DERIVATIVE

        !"Inverter1: Generate carrier triangle,tt=0 is at positive side maximum"
```

```
        tt = mod(t,Ttri)        !"modulus,remainder t/Ttri,position on tri"
        IF (tt .LT. Tover2) Then        !"Before zero crossing"
            Vtri = Vtripk -slope*tt
        ELSE
            Vtri = -Vtripk + slope*(tt-Tover2)
        ENDIF

! The synch-rf angle is theta.  Theta=w*t
        theta1=INTEG(we1,0.0)
        theta2=theta1
        theta3=theta1

! Sharing stuff
Rpwr1=3.0/2.0*(Vcq1*I2q1 + Vcd1*I2d1)
Rpwr2=3.0/2.0*(Vcq2*I2q2 + Vcd2*I2d2)
Rpwr3=3.0/2.0*(Vcq3*I2q3 + Vcd3*I2d3)

!Prevent negative powers from messing up the average
If(Rpwr1 .LT. 0.0)Then
        TRpwr1=0.0
Else
        TRpwr1=Rpwr1
EndIf
If(Rpwr2 .LT. 0.0)Then
        TRpwr2=0.0
Else
        TRpwr2=Rpwr2
EndIf
If(Rpwr3 .LT. 0.0)Then
        TRpwr3=0.0
Else
        TRpwr3=Rpwr3
EndIf

Qpwr1 = 3.0/2.0 *(Vcq1*I2d1-Vcd1*I2q1)
Qpwr2 = 3.0/2.0 *(Vcq2*I2d2-Vcd2*I2q2)
Qpwr3 = 3.0/2.0 *(Vcq3*I2d3-Vcd3*I2q3)

Iline1=sqrt(I2q1**2 + I2d1**2)
Iline2=sqrt(I2q2**2 + I2d2**2)
Iline3=sqrt(I2q3**2 + I2d3**2)

! Assume 15Kw inverters so max allowed load difference is 8%
```

```
! Real Power Sharing
IF (t .GE. sharetimeR) Then
   sharetimeR= t + .01 !reset to next sample time

 ! Calculate desired droop location
        Desired1= 455 - TRpwr1*10.0/15000.0
        Desired2= 455 - TRpwr2*10.0/15000.0
        Desired3= 455 - TRpwr3*10.0/15000.0
        Vtemp1=Vqcom1
        Vtemp2=Vqcom2
        Vtemp3=Vqcom3
 IF((Desired1.GT.Vqcom1).AND.((Desired1-Vqcom1).GT.(.05)))Then
        Vqcom1=Vtemp1 + Vstep*(Desired1-Vqcom1)*onoff1
 ElseIf((Desired1.LT.Vqcom1).AND.((Vqcom1-Desired1).GT.(.05)))Then
        Vqcom1=Vtemp1 - Vstep*(Vqcom1-Desired1)*onoff1
 Endif

 IF((Desired2.GT.Vqcom2).AND.((Desired2-Vqcom2).GT.(.05)))Then
        Vqcom2=Vtemp2 + Vstep*(Desired2-Vqcom2)*onoff2
 ElseIf((Desired2.LT.Vqcom2).AND.((Vqcom2-Desired2).GT.(.05)))Then
        Vqcom2=Vtemp2 - Vstep*(Vqcom2-Desired2)*onoff2       ! power down
 Endif
 IF((Desired3.GT.Vqcom3).AND.((Desired3-Vqcom3).GT.(.05)))Then
        Vqcom3=Vtemp3 + Vstep*(Desired3-Vqcom3)*onoff3
 ElseIf((Desired3.LT.Vqcom3).AND.((Vqcom3-Desired3).GT.(.05)))Then
        Vqcom3=Vtemp3 - Vstep*(Vqcom3-Desired3)*onoff3       ! power down
 Endif
Endif

        !"Generate Control Signals. Based on load voltage errors, and filter currents
                !Load voltage
                vqerr1=Vqcom1 - Vcq1
                vderr1=Vdcom1 - Vcd1
                vqerr2=Vqcom2 - Vcq2
                vderr2=Vdcom2 - Vcd2
                vqerr3=Vqcom3 - Vcq3
                vderr3=Vdcom3 - Vcd3

                !PI voltage controller constants
                kpvq=9.262042
                kpvd=1.7747558e+01
                kivq=4.05144e+03
                kivd=4.05144e+03
```

```
!Rename the errors as derivatives (d) and find states
dx91=vqerr1
dx101=vderr1
x91=INTEG(dx91,0.0)
x101=INTEG(dx101,0.0)
icapq1=kpvq*dx91 + kivq*x91
icapd1=kpvd*dx101 + kivd*x101
dx92=vqerr2
dx102=vderr2
x92=INTEG(dx92,0.0)
x102=INTEG(dx102,0.0)
icapq2=kpvq*dx92 + kivq*x92
icapd2=kpvd*dx102 + kivd*x102
dx93=vqerr3
dx103=vderr3
x93=INTEG(dx93,0.0)
x103=INTEG(dx103,0.0)
icapq3=kpvq*dx93 + kivq*x93
icapd3=kpvd*dx103 + kivd*x103

!feedforward action from desired filter capacitor currents
icqcom1=icapq1 + we1*CF*Vdcom1
icdcom1=icapd1 - we1*CF*Vqcom1
icqcom2=icapq2 + we2*CF*Vdcom2
icdcom2=icapd2 - we2*CF*Vqcom2
icqcom3=icapq3 + we3*CF*Vdcom3
icdcom3=icapd3 - we3*CF*Vqcom3

!desired inductor current in the filter
ifqcom1=icqcom1 + I2q1
ifdcom1=icdcom1 + I2d1
ifqcom2=icqcom2 + I2q2
ifdcom2=icdcom2 + I2d2
ifqcom3=icqcom3 + I2q3
ifdcom3=icdcom3 + I2d3

!use the error in the filter currents for normal control
cqerr1=ifqcom1-Ifq1
cderr1=ifdcom1-Ifd1
dx111=cqerr1
dx121=cderr1
cqerr2=ifqcom2-Ifq2
cderr2=ifdcom2-Ifd2
dx112=cqerr2*onoff22
```

```
dx122=cderr2*onoff22
cqerr3=ifqcom3-Ifq3
cderr3=ifdcom3-Ifd3
dx113=cqerr3*onoff23
dx123=cderr3*onoff23

!PI current controller contstants
kpcq=3.1640491671e-01
kpcd=3.588323233e-01
kicq=5.0642793e+02
kicd=5.0642793e+02

x111=INTEG(dx111,0.0)
x121=INTEG(dx121,0.0)
x112=INTEG(dx112,0.0)
x122=INTEG(dx122,0.0)
x113=INTEG(dx113,0.0)
x123=INTEG(dx123,0.0)

! Feedback correction for the perturbations
vfbq1=kpcq*dx111 + kicq*x111
vfbd1=kpcd*dx121 + kicd*x121
vfbq2=kpcq*dx112 + kicq*x112
vfbd2=kpcd*dx122 + kicd*x122
vfbq3=kpcq*dx113 + kicq*x113
vfbd3=kpcd*dx123 + kicd*x123

! Commanded Inverter voltages
Vqncom1=Vcq1 + we1*LF*ifdcom1 + vfbq1        !note ifd not ifq
Vdncom1=Vcd1 - we1*LF*ifqcom1 + vfbd1
Vqncom2=Vcq2 + we2*LF*ifdcom2 + vfbq2        !note ifd not ifq
Vdncom2=Vcd2 - we2*LF*ifqcom2 + vfbd2
Vqncom3=Vcq3 + we3*LF*ifdcom3 + vfbq3        !note ifd not ifq
Vdncom3=Vcd3 - we3*LF*ifqcom3 + vfbd3

! Transform the generated commanded voltages to abc quantities (gain = 1)
    ! use in the sine-triangle switching scheme !
    va1 = Vqncom1*cos(theta1) + Vdncom1*sin(theta1)
    vb1 = Vqncom1*cos(theta1-rad120) + Vdncom1*sin(theta1-rad120)
    vc1 = -va1 -vb1          ! balanced loads
    va2 = Vqncom2*cos(theta2) + Vdncom2*sin(theta2)
    vb2 = Vqncom2*cos(theta2-rad120)+Vdncom2*sin(theta2-rad120)
    vc2 = -va2 -vb2          ! balanced loads
    va3 = Vqncom3*cos(theta3) + Vdncom3*sin(theta3)
```

```
        vb3 = Vqncom3*cos(theta3-rad120)+Vdncom3*sin(theta3-rad120)
        vc3 = -va3 -vb3         ! balanced loads

    !"Control inverter switches to change state at crossing times of Va and Vtri"
        SCHEDULE hia1 .XN. (Vtri - va1)
        SCHEDULE loa1 .XP. (Vtri - va1)
        SCHEDULE hib1 .XN. (Vtri - vb1)
        SCHEDULE lob1 .XP. (Vtri - vb1)
        SCHEDULE hic1 .XN. (Vtri - vc1)
        SCHEDULE loc1 .XP. (Vtri - vc1)
        SCHEDULE hia2 .XN. (Vtri - va2)
        SCHEDULE loa2 .XP. (Vtri - va2)
        SCHEDULE hib2 .XN. (Vtri - vb2)
        SCHEDULE lob2 .XP. (Vtri - vb2)
        SCHEDULE hic2 .XN. (Vtri - vc2)
        SCHEDULE loc2 .XP. (Vtri - vc2)
        SCHEDULE hia3 .XN. (Vtri - va3)
        SCHEDULE loa3 .XP. (Vtri - va3)
        SCHEDULE hib3 .XN. (Vtri - vb3)
        SCHEDULE lob3 .XP. (Vtri - vb3)
        SCHEDULE hic3 .XN. (Vtri - vc3)
        SCHEDULE loc3 .XP. (Vtri - vc3)

!"Sets the voltage on the phase V_p, p is the DC input common, WYE Connected"
    !"Procedural has the outputs then the inputs in a list."
        PROCEDURAL(vas1,vbs1,vcs1=SA1,SB1,SC1,vi)

        !"Assume the switch voltage is zero unless the logic proves us wrong"
            Vap1=0.0
            Vbp1=0.0
            Vcp1=0.0

        !"Status of top switch determines voltage applied to the phase"
            IF (SA1) THEN
                Vap1 = vi
            ENDIF
            IF (SB1) THEN
                Vbp1 = vi
            ENDIF
            IF (SC1) THEN
                Vcp1 = vi
            ENDIF

        ! establish the phase voltages assuming a balance three phase load
```

```
        Vnp1 = (Vap1 + Vbp1 + Vcp1)/3.0
        vas1 = Vap1 - Vnp1
        vbs1 = Vbp1 - Vnp1
        vcs1 = Vcp1 - Vnp1

END !"procedural"

PROCEDURAL(vas2,vbs2,vcs2=SA2,SB2,SC2,vi)

!"Assume the switch voltage is zero unless the logic proves us wrong"
        Vap2=0.0
        Vbp2=0.0
        Vcp2=0.0

!"Status of top switch determines voltage applied to the phase"
        IF (SA2) THEN
                Vap2 = vi
        ENDIF
        IF (SB2) THEN
                Vbp2 = vi
        ENDIF
        IF (SC2) THEN
                Vcp2 = vi
        ENDIF

! establish the phase voltages assuming a balance three phase load
        Vnp2 = (Vap2 + Vbp2 + Vcp2)/3.0
        vas2 = Vap2 - Vnp2
        vbs2 = Vbp2 - Vnp2
        vcs2 = Vcp2 - Vnp2

END !"procedural"

PROCEDURAL(vas3,vbs3,vcs3=SA3,SB3,SC3,vi)

!"Assume the switch voltage is zero unless the logic proves us wrong"
        Vap3=0.0
        Vbp3=0.0
        Vcp3=0.0

!"Status of top switch determines voltage applied to the phase"
        IF (SA3) THEN
                Vap3 = vi
        ENDIF
```

```
                IF (SB3) THEN
                        Vbp3 = vi
                ENDIF
                IF (SC3) THEN
                        Vcp3 = vi
                ENDIF

        ! establish the phase voltages assuming a balance three phase load
                Vnp3 = (Vap3 + Vbp3 + Vcp3)/3.0
                vas3 = Vap3 - Vnp3
                vbs3 = Vbp3 - Vnp3
                vcs3 = Vcp3 - Vnp3

        END !"procedural"

    ! Unit 1 inverter model
VQn1 = 2.0*(vas1*cos(theta1)+vbs1*cos(theta1-rad120)+vcs1*cos(theta1+rad120))/3.0
VDn1 = 2.0*(vas1*sin(theta1)+vbs1*sin(theta1-rad120)+vcs1*sin(theta1+rad120))/3.0
  dIfq1=-we1*Ifd1 + 1./Lf*(VQn1-Vcq1-Ifq1*RF)
    dVcq1=-we1*Vcd1 + 1./Cf*(Ifq1-I2q1)
    dI2q1=-we1*I2d1 + 1./L2*(Vcq1-Vlq1-I2q1*R)
    dVlq1=-we1*Vld1 + 1./Cl*(I2q1+I2q2+I2q3-Ilq1)
    dIlq1=-we1*Ild1 + 1./Ll*(Vlq1-Ilq1*RL)

    dIfd1=we1*Ifq1 + 1./Lf*(VDn1-Vcd1-Ifd1*RF)
    dVcd1=we1*Vcq1 + 1./Cf*(Ifd1-I2d1)
    dI2d1=we1*I2q1 + 1./L2*(Vcd1-Vld1-I2d1*R)
    dVld1=we1*Vlq1 + 1./Cl*(I2d1+I2d2+I2d3-Ild1) !This Vl and Il are used by all
    dIld1=we1*Ilq1 + 1./Ll*(Vld1-Ild1*RL)

    Ifq1=integ(dIfq1,0.0)
    Vcq1=integ(dVcq1,0.0)
    I2q1=integ(dI2q1,0.0)
    Vlq1=integ(dVlq1,0.0)
    Ilq1=integ(dIlq1,0.0)
    Ifd1=integ(dIfd1,0.0)
    Vcd1=integ(dVcd1,0.0)
    I2d1=integ(dI2d1,0.0)
    Vld1=integ(dVld1,0.0)
    Ild1=integ(dIld1,0.0)

    ! UNIT 2 inverter model
VQn2 = 2.0*(vas2*cos(theta2)+vbs2*cos(theta2-rad120)+vcs2*cos(theta2+rad120))/3.0
VDn2 = 2.0*(vas2*sin(theta2)+vbs2*sin(theta2-rad120)+vcs2*sin(theta2+rad120))/3.0
```

```
      dIfq2=-we2*Ifd2 + 1./Lf*(VQn2-Vcq2-Ifq2*RF)
        dVcq2=-we2*Vcd2 + 1./Cf*(Ifq2-I2q2)
        dI2q2=-we2*I2d2 + 1./L22*(Vcq2-Vlq1-I2q2*R2)*onoff22
        dIfd2=we2*Ifq2 + 1./Lf*(VDn2-Vcd2-Ifd2*RF)
        dVcd2=we2*Vcq2 + 1./Cf*(Ifd2-I2d2)
        dI2d2=we2*I2q2 + 1./L22*(Vcd2-Vld1-I2d2*R2)*onoff22

        Ifq2=integ(dIfq2,0.0)
        Vcq2=integ(dVcq2,0.0)
        I2q2=integ(dI2q2,0.0)
        Ifd2=integ(dIfd2,0.0)
        Vcd2=integ(dVcd2,0.0)
        I2d2=integ(dI2d2,0.0)

        ! UNIT 3 inverter model
VQn3 = 2.0*(vas3*cos(theta3)+vbs3*cos(theta3-rad120)+vcs3*cos(theta3+rad120))/3.0
VDn3 = 2.0*(vas3*sin(theta3)+vbs3*sin(theta3-rad120)+vcs3*sin(theta3+rad120))/3.0
      dIfq3=-we3*Ifd3 + 1./Lf*(VQn3-Vcq3-Ifq3*RF)
        dVcq3=-we3*Vcd3 + 1./Cf*(Ifq3-I2q3)
        dI2q3=-we3*I2d3 + 1./L23*(Vcq3-Vlq1-I2q3*R3)*onoff23
        dIfd3=we3*Ifq3 + 1./Lf*(VDn3-Vcd3-Ifd3*RF)
        dVcd3=we3*Vcq3 + 1./Cf*(Ifd3-I2d3)
        dI2d3=we3*I2q3 + 1./L23*(Vcd3-Vld1-I2d3*R3)*onoff23

        Ifq3=integ(dIfq3,0.0)
        Vcq3=integ(dVcq3,0.0)
        I2q3=integ(dI2q3,0.0)
        Ifd3=integ(dIfd3,0.0)
        Vcd3=integ(dVcd3,0.0)
        I2d3=integ(dI2d3,0.0)

!******** The outputs of interest *****************
        VLoad= Vlq1*cos(theta1) + Vld1*sin(theta1)
        Iload= Ilq1*cos(theta1) + Ild1*sin(theta1)
        Imax=sqrt(Ilq1**2 + Ild1**2)
        Vmax=sqrt(Vlq1**2 + Ild1**2)
!*************************************************
        END !"derivative"

        DISCRETE  hia1        !"Evaluate Scheduled events
                SA1 = .TRUE.
        END
        DISCRETE loa1
                SA1 = .FALSE.
```

```
END
DISCRETE  hib1
        SB1 = .TRUE.
END
DISCRETE lob1
        SB1 = .FALSE.
END
DISCRETE  hic1
        SC1 = .TRUE.
END
DISCRETE loc1
        SC1 = .FALSE.
END                     !"last of the discretes for Inverter1"

DISCRETE  hia2          !"Evaluate Scheduled events
        SA2 = .TRUE.
END
DISCRETE loa2
        SA2 = .FALSE.
END
DISCRETE  hib2
        SB2 = .TRUE.
END
DISCRETE lob2
        SB2 = .FALSE.
END
DISCRETE  hic2
        SC2 = .TRUE.
END
DISCRETE loc2
        SC2 = .FALSE.
END                     !"last of the discretes for Inverter2"

DISCRETE  hia3          !"Evaluate Scheduled events
        SA3 = .TRUE.
END
DISCRETE loa3
        SA3 = .FALSE.
END
DISCRETE  hib3
        SB3 = .TRUE.
END
DISCRETE lob3
        SB3 = .FALSE.
```

```
        END
        DISCRETE  hic3
                SC3 = .TRUE.
        END
        DISCRETE loc3
                SC3 = .FALSE.
        END                     !"last of the discretes for Inverter3"

END !"Dynamic"

!" no TERMINAL"

END !"Program


!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! share2.cmd preference File

s strplt = .t.          ! "one variable per x-axis"
s calplt = .f.          ! "not continuous plot"

s devplt = 1            !"1=screen output"

s ppoplt = .f.                  ! "true rotates plot 90 deg for pen plotters"

!"together these set the plot aspect ratio"
s xinspl = 7                    ! "# of blocks for x axis"
s yinspl = 8                    ! "# of blocks for y axis"

s weditg = .f.                  ! "false suppresses data write"
                                ! "each time SCHEDULE occurs"

s nrwitg = .f.                  ! "true enables accumulation of data"
                                ! "after a CONTIN"
s alcplt = .f.                  ! "plot color not on"

prepare t,Vload,Iload,Iline1,Iline2,Rpwr1,Rpwr2,Qpwr1,Qpwr2
prepare Rpwr3,Qpwr3

proced p1                       !"creates script called p1, run from acsl
        plot  Vload
end

proced p2
```

```
        plot Iload
end
proced p3
        plot Rpwr1
end
proced p4
        plot Rpwr2
end
proced p5
        plot Rpwr3
end
proced p6
        plot Qpwr1
end
proced p7
        plot Qpwr2
end
proced p8
        plot Qpwr3
end
proced p8
        plot Imax
end
proced nr
        s nrwitg=.t.
end
proced yr
        s nrwitg=.f.
end
proced sv
        save /events /file='saveshare3'
end
proced rt
        s nrwitg=.f.            ! gets rid of historic data
        s t=0.0                 ! resets the time
        s tstop=.0001           ! tiny run to initialize
        start                   ! initializes all the variables
        restore /events /file='saveshare3' !restores the old data
        s t=0.0
        s tstop=.02
        cont
        s nrwitg=.t.
        s tstop=.05
        s sharetimer=t
```

```
end
proced share
        s onoff1=1.0
        s onoff2=1.0
        s sharetimer=t
end
proced ml
        matlab vtemp1
end
proced vars
        d rl, ll
        d r,r2,r3
        d rpwr1,rpwr2,rpwr3
        d qpwr1,qpwr2,qpwr3
        d Iline1,Iline2,Iline3
        d desired1,desired2,desired3
        d vqcom1,vqcom2,vqcom3
        d vmax
end

!" /xhi= sets max x for plot (/xlo)"
```

APPENDIX B: ACSL FREQUENCY DROOP CODE

```
PROGRAM   ! Freq2.CSL
! Test FREQUENCY DROOP SHARING

INITIAL
      MAXTERVAL maxt = 1.0e-6          !"max integration step size norm=e-5"
      MINTERVAL mint = 1.0e-8  !"min time step for int algorithm norm=e-7
      CINTERVAL cint = 1.0e-4    ! How often data written to disk
(small=smooth,slow)
      ALGORITHM ialg = 5               !"4rth order runga kutta"
      NSTEPS nstp = 1                  !" 1 = int step size controlled by maxt"
      CONSTANT tstop = 0.1             !"stop point for integration"
      PARAMETER (twopi = 6.283185307)        !" 2 times pi = omega times period"
      CONSTANT rad120 = 2.094395       !" 120 degrees in radians"
      CONSTANT NL1 =377.0              ! 60 hz system
      CONSTANT NL2 =377.0              ! 60 hz system
      CONSTANT we1 =377.0              ! 60 hz system
      CONSTANT we2 =377.0              ! 60 hz system

      !"DC Source Constants"
      CONSTANT vi = 950.0              !" first inverter DC input"

!"Switch Description, only describe top switches, bottom complimentary, 3 phase"

      LOGICAL SA1,SB1,SC1              !"Inverter1/2 switch pairs"
      SA1 = .FALSE.                    !"Top switch open, bottom switch shut"
      SB1 = .FALSE.
      SC1 = .FALSE.
      LOGICAL SA2,SB2,SC2
      SA2 = .FALSE.                    !"Top switch open, bottom switch shut"
      SB2 = .FALSE.
      SC2 = .FALSE.

      !"Carrier Triangle Constants, symetric, bipolar triangle"
CONSTANT ftriang = 10800.0       !"triangle freq in Hz, want this > 9*fsin" (10K)
      Vtripk = vi/2.0            !"peak triangle amplitude"
      Ttri = 1.0/ftriang         !"triangle period in sec"
      Tover2 = Ttri/2.0          !"triangle half period"
      slope = 4.0*Vtripk*ftriang !"slope of the triangle,1/4 of period"

      CONSTANT LF=5.626977e-05       !"inverter one output inductor(.004)
      CONSTANT CF=1.1253954e-02
```

```
        !"bank capacitor value (e-4)
      CONSTANT RF=.1                    ! filter resistance
!
      CONSTANT R=.07                    !"series line resistance, source 1 (.0001)
      CONSTANT L2=1.0e-6                ! series line inductance
      Constant R2=.035                  ! inv2
      Constant L22=1.0e-6


      !"Phase load constants"
      CONSTANT RL=20.0                  !"resistor load in each leg"
      CONSTANT CL=1.0e-5                !"load capacitor value (e-6)
      CONSTANT LL=.021                  !"0.002 inductor in each load"

      ! sharing values
      Constant Vqcom1=450.0             ! No-Load commanded voltage
      Constant Vdcom1=0.0
      Constant Vqcom2=450.0
      Constant Vdcom2=0.0

      Constant dslope1=5.0      !.05
      Constant dslope2=5.0      !.05
      Constant fstep=.02        !1.0
      Constant onoff=1.0        !0.0 sets freq droop off
      Constant sharetimer=.05
      Constant update=.01

END   !"initial"

DYNAMIC
      !"This section is executed each cint seconds"
      !"This statement stops simulation at tstop seconds"
      TERMT (t .GE. (tstop-0.5*cint))

      DERIVATIVE

      !"Inverter1: Generate carrier triangle,tt=0 is at positive side maximum"
            tt = mod(t,Ttri)          !"modulus,remainder t/Ttri,position on tri"
            IF (tt .LT. Tover2) Then      !"Before zero crossing"
                  Vtri = Vtripk -slope*tt
            ELSE
                  Vtri = -Vtripk + slope*(tt-Tover2)
            ENDIF
```

```
! The synch-rf angle is theta.  Theta=w*t
        theta1=INTEG(we1,0.0)
        theta2=INTEG(we2,0.0)

! Sharing stuff
Rpwr1=3.0/2.0*(Vcq1*I2q1 + Vcd1*I2d1)
Rpwr2=3.0/2.0*(Vcq2*I2q2 + Vcd2*I2d2)

we1=NL1-(dslope1*Rpwr1/15000.0)*onoff
we2=NL2-(dslope2*Rpwr2/15000.0)*onoff

Schedule frecover .XP. (t-sharetimer)


Iline1=sqrt(I2q1**2 + I2d1**2)
Iline2=sqrt(I2q2**2 + I2d2**2)


!"Generate Control Signals. Based on load voltage errors, and filter currents
        !Load voltage
        vqerr1=Vqcom1 - Vcq1
        vderr1=Vdcom1 - Vcd1
        vqerr2=Vqcom2 - Vcq2
        vderr2=Vdcom2 - Vcd2

        !PI voltage controller constants
        kpvq=9.262042
        kpvd=1.7747558e+01
        kivq=4.05144e+03
        kivd=4.05144e+03

        !Rename the errors as derivatives (d) and find states
        dx91=vqerr1
        dx101=vderr1
        x91=INTEG(dx91,0.0)
        x101=INTEG(dx101,0.0)
        icapq1=kpvq*dx91 + kivq*x91
        icapd1=kpvd*dx101 + kivd*x101
        dx92=vqerr2
        dx102=vderr2
        x92=INTEG(dx92,0.0)
        x102=INTEG(dx102,0.0)
        icapq2=kpvq*dx92 + kivq*x92
        icapd2=kpvd*dx102 + kivd*x102
```

```
!feedforward action from desired filter capacitor currents
icqcom1=icapq1 + we1*CF*Vdcom1
icdcom1=icapd1 - we1*CF*Vqcom1
icqcom2=icapq2 + we2*CF*Vdcom2
icdcom2=icapd2 - we2*CF*Vqcom2

!desired inductor current in the filter
ifqcom1=icqcom1 + I2q1
ifdcom1=icdcom1 + I2d1
!!
ifqcom2=icqcom2 + I2q2
ifdcom2=icdcom2 + I2d2

!use the error in the filter currents for normal control
cqerr1=ifqcom1-Ifq1
cderr1=ifdcom1-Ifd1
dx111=cqerr1
dx121=cderr1
cqerr2=ifqcom2-Ifq2
cderr2=ifdcom2-Ifd2
dx112=cqerr2
dx122=cderr2

!PI current controller contstants
kpcq=3.1640491671e-01
kpcd=3.588323233e-01
kicq=5.0642793e+02
kicd=5.0642793e+02

x111=INTEG(dx111,0.0)
x121=INTEG(dx121,0.0)
x112=INTEG(dx112,0.0)
x122=INTEG(dx122,0.0)

! Feedback correction for the perturbations
vfbq1=kpcq*dx111 + kicq*x111
vfbd1=kpcd*dx121 + kicd*x121
vfbq2=kpcq*dx112 + kicq*x112
vfbd2=kpcd*dx122 + kicd*x122

! Commanded Inverter voltages
Vqncom1=Vcq1 + we1*LF*ifdcom1 + vfbq1        !note ifd not ifq
Vdncom1=Vcd1 - we1*LF*ifqcom1 + vfbd1
```

```
            Vqncom2=Vcq2 + we2*LF*ifdcom2 + vfbq2        !note ifd not ifq
            Vdncom2=Vcd2 - we2*LF*ifqcom2 + vfbd2

            ! Transform the generated commanded voltages to abc quantities (gain =1)
            ! use in the sine-triangle switching scheme !
            va1 = Vqncom1*cos(theta1) + Vdncom1*sin(theta1)
            vb1 = Vqncom1*cos(theta1-rad120) + Vdncom1*sin(theta1-rad120)
            vc1 = -va1 -vb1        ! balanced loads
!
            va2 = Vqncom2*cos(theta2) + Vdncom2*sin(theta2)
            vb2 = Vqncom2*cos(theta2-rad120)+Vdncom2*sin(theta2-rad120)
            vc2 = -va2 -vb2        ! balanced loads

        !"Control inverter switches to change state at crossing times of Va and Vtri"
            SCHEDULE hia1 .XN. (Vtri - va1)
            SCHEDULE loa1 .XP. (Vtri - va1)
            SCHEDULE hib1 .XN. (Vtri - vb1)
            SCHEDULE lob1 .XP. (Vtri - vb1)
            SCHEDULE hic1 .XN. (Vtri - vc1)
            SCHEDULE loc1 .XP. (Vtri - vc1)
            SCHEDULE hia2 .XN. (Vtri - va2)
            SCHEDULE loa2 .XP. (Vtri - va2)
            SCHEDULE hib2 .XN. (Vtri - vb2)
            SCHEDULE lob2 .XP. (Vtri - vb2)
            SCHEDULE hic2 .XN. (Vtri - vc2)
            SCHEDULE loc2 .XP. (Vtri - vc2)

        !"Sets the voltage on the phase V_p, p is the DC input common, WYE
Connected"
        !"Procedural has the outputs then the inputs in a list."
            PROCEDURAL(vas1,vbs1,vcs1=SA1,SB1,SC1,vi)

            !"Assume the switch voltage is zero unless the logic proves us wrong"
                Vap1=0.0
                Vbp1=0.0
                Vcp1=0.0

            !"Status of top switch determines voltage applied to the phase"
                IF (SA1) THEN
                    Vap1 = vi
                ENDIF
                IF (SB1) THEN
                    Vbp1 = vi
                ENDIF
```

```
            IF (SC1) THEN
                Vcp1 = vi
            ENDIF

        ! establish the phase voltages assuming a balance three phase load
            Vnp1 = (Vap1 + Vbp1 + Vcp1)/3.0
            vas1 = Vap1 - Vnp1
            vbs1 = Vbp1 - Vnp1
            vcs1 = Vcp1 - Vnp1

        END !"procedural"

        PROCEDURAL(vas2,vbs2,vcs2=SA2,SB2,SC2,vi)

        !"Assume the switch voltage is zero unless the logic proves us wrong"
            Vap2=0.0
            Vbp2=0.0
            Vcp2=0.0

        !"Status of top switch determines voltage applied to the phase"
            IF (SA2) THEN
                Vap2 = vi
            ENDIF
            IF (SB2) THEN
                Vbp2 = vi
            ENDIF
            IF (SC2) THEN
                Vcp2 = vi
            ENDIF

        ! establish the phase voltages assuming a balance three phase load
            Vnp2 = (Vap2 + Vbp2 + Vcp2)/3.0
            vas2 = Vap2 - Vnp2
            vbs2 = Vbp2 - Vnp2
            vcs2 = Vcp2 - Vnp2

        END !"procedural"

    ! Unit 1 inverter model
VQn1 = 2.0*(vas1*cos(theta1)+vbs1*cos(theta1-rad120)+vcs1*cos(theta1+rad120))/3.0
VDn1 = 2.0*(vas1*sin(theta1)+vbs1*sin(theta1-rad120)+vcs1*sin(theta1+rad120))/3.0
  dIfq1=-we1*Ifd1 + 1./Lf*(VQn1-Vcq1-Ifq1*RF)
    dVcq1=-we1*Vcd1 + 1./Cf*(Ifq1-I2q1)
    dI2q1=-we1*I2d1 + 1./L2*(Vcq1-Vlq1-I2q1*R)
```

```
        dVlq1=-we1*Vld1 + 1./Cl*(I2q1+TI2q2-Ilq1)
        dIlq1=-we1*Ild1 + 1./Ll*(Vlq1-Ilq1*RL)

        dIfd1=we1*Ifq1 + 1./Lf*(VDn1-Vcd1-Ifd1*RF)
        dVcd1=we1*Vcq1 + 1./Cf*(Ifd1-I2d1)
        dI2d1=we1*I2q1 + 1./L2*(Vcd1-Vld1-I2d1*R)
        dVld1=we1*Vlq1 + 1./Cl*(I2d1+TI2d2-Ild1) !This Vl and Il are used by all
        dIld1=we1*Ilq1 + 1./Ll*(Vld1-Ild1*RL)

        Ifq1=integ(dIfq1,0.0)
        Vcq1=integ(dVcq1,0.0)
        I2q1=integ(dI2q1,0.0)
        Vlq1=integ(dVlq1,0.0)
        Ilq1=integ(dIlq1,0.0)
        Ifd1=integ(dIfd1,0.0)
        Vcd1=integ(dVcd1,0.0)
        I2d1=integ(dI2d1,0.0)
        Vld1=integ(dVld1,0.0)
        Ild1=integ(dIld1,0.0)
!!
        TVld1=Vlq1*sin(theta2-theta1)+Vld1*cos(theta2-theta1) !needed for unit 2`s RF
        TVlq1=Vlq1*cos(theta2-theta1)-Vlq1*sin(theta2-theta1)

        ! UNIT 2 inverter model
VQn2 = 2.0*(vas2*cos(theta2)+vbs2*cos(theta2-rad120)+vcs2*cos(theta2+rad120))/3.0
VDn2 = 2.0*(vas2*sin(theta2)+vbs2*sin(theta2-rad120)+vcs2*sin(theta2+rad120))/3.0
      dIfq2=-we2*Ifd2 + 1./Lf*(VQn2-Vcq2-Ifq2*RF)
        dVcq2=-we2*Vcd2 + 1./Cf*(Ifq2-I2q2)
        dI2q2=-we2*I2d2 + 1./L22*(Vcq2-TVlq1-I2q2*R2)
        dIfd2=we2*Ifq2 + 1./Lf*(VDn2-Vcd2-Ifd2*RF)
        dVcd2=we2*Vcq2 + 1./Cf*(Ifd2-I2d2)
        dI2d2=we2*I2q2 + 1./L22*(Vcd2-TVld1-I2d2*R2)

        Ifq2=integ(dIfq2,0.0)
        Vcq2=integ(dVcq2,0.0)
        I2q2=integ(dI2q2,0.0)
        Ifd2=integ(dIfd2,0.0)
        Vcd2=integ(dVcd2,0.0)
        I2d2=integ(dI2d2,0.0)

        TI2q2=I2q2*cos(theta1-theta2)-I2d2*sin(theta1-theta2) !needed for unit 1's RF
        TI2d2=I2q2*sin(theta1-theta2)+I2d2*cos(theta1-theta2)

!******** The outputs of interest **********
```

```
      VLoad= Vlq1*cos(theta1) + Vld1*sin(theta1)
      Iload= Ilq1*cos(theta1) + Ild1*sin(theta1)
      Imax=sqrt(Ilq1**2 + Ild1**2)
      Vmax=sqrt(Vlq1**2 + Ild1**2)
!*************************************************
      END !"derivative"

      DISCRETE  hia1        !"Evaluate Scheduled events
            SA1 = .TRUE.
      END
      DISCRETE loa1
            SA1 = .FALSE.
      END
      DISCRETE  hib1
            SB1 = .TRUE.
      END
      DISCRETE lob1
            SB1 = .FALSE.
      END
      DISCRETE  hic1
            SC1 = .TRUE.
      END
      DISCRETE loc1
            SC1 = .FALSE.
      END                   !"last of the discretes for Inverter1"

      DISCRETE  hia2        !"Evaluate Scheduled events
            SA2 = .TRUE.
      END
      DISCRETE loa2
            SA2 = .FALSE.
      END
      DISCRETE  hib2
            SB2 = .TRUE.
      END
      DISCRETE lob2
            SB2 = .FALSE.
      END
      DISCRETE  hic2
            SC2 = .TRUE.
      END
      DISCRETE loc2
            SC2 = .FALSE.
      END                   !"last of the discretes for Inverter2"
```

```
        Discrete frecover
                sharetimeR= t + update          !reset to next sample time
                tempf1=NL1
                tempf2=NL2
                NL1= tempf1 + fstep*(377-we1)
                NL2 = tempf2 + fstep*(377-we2)
        End

END !"Dynamic"

!" no TERMINAL"

END !"Program

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! phase2.cmd preference File

s strplt = .t.          ! "one variable per x-axis"
s calplt = .f.          ! "not continuous plot"

s devplt = 1            !"1=screen output"

s ppoplt = .f.                  ! "true rotates plot 90 deg for pen plotters"

!"together these set the plot aspect ratio"
s xinspl = 7                    ! "# of blocks for x axis"
s yinspl = 8                    ! "# of blocks for y axis"

s weditg = .f.                  ! "false suppresses data write"
                                ! "each time SCHEDULE occurs"

s nrwitg = .f.                  ! "true enables accumulation of data"
                                ! "after a CONTIN"
s alcplt = .f.                  ! "plot color not on"

prepare t,Vload,Iload,Iline1,Iline2,Rpwr1,Rpwr2
prepare we1,we2,NL1, NL2

proced p1                       !"creates script called p1, run from acsl
        plot  Vload
end

proced p2
        plot Iload
```

```
end
proced p3
        plot Rpwr1,Rpwr2
end
proced p4
        plot we1,we2
end
proced p5
        plot NL1,NL2
end
proced nr
        s nrwitg=.t.
end
proced sv
        save /events /file='savefreq1'
end
proced rt
        restore /events /file='savefreq1'
        s nrwitg=.f.
end
proced start2
        s onoff=1.0
        s state7=0.0
end
proced vars
        d rpwr1,rpwr2
        d we1,we2
        d r,r2
        d Iline1,Iline2
        d vmax
end
proced up
        s rl=10.0
        s ll=.0043
end
proced dwn
        s rl=20.0
        s ll=.021
end

!" /xhi= sets max x for plot (/xlo)"
```

LIST OF REFERENCES

[1] Parker, D.S., Hodge, C.G., "Electric Warship," *Power Engineering Journal*, Vol. 12, No. 1, February 1998.

[2] Kawabata, T., Higashino S., "Parallel Operation of Voltage Source Inverters," *IEEE Transactions on Industry Applications*, Vol. 24, No. 2, March/April 1988.

[3] Ryan, M., Lorenz, R., "A High Performance Sine Wave Inverter Controller with Capacitor Current Feedback and "Back-EMF" Decoupling," Proc. of the 26th Annual IEEE Power Electronics Specialists Conference, Vol. 1, June 1995.

[4] Divan, D.M and Chandorkar, M., *Method and apparatus for de-centralized signal frequency restoration in a distributed UPS system*, U.S. Patent 5,596,492, 21 January 1997.

[5] Chandorkar, M., Divan, D., Adapa, R., "Control of Parallel Connected Inverters in Stand-Alone AC Supply Systems," *IEEE Transactions on Industry Applications*, v. 29, pp. 136-143, February 1993.

[6] Perreault, D.J., Selders Jr. R.L., and Kassakian, J.G., "Frequency-Based Current-Sharing Techniques for Parallel Power Converters," *IEEE Transactions on Power Electronics*, v. 13, no. 4, pp. 626-634, 1998.

[7] Mohan N., Undeland, T.M., Robbins, W.P., *Power Electronics Converters, Applications, and Design*, John Wiley & Sons, Inc., 1989.

[8] Krein, P.T., *Elements of Power Electronics*, Oxford University Press, Inc., 1998.

[9] Fisher, M.J., *Power Electronics*, PWS-KENT Publishing Company, 1991.

[10] Department of Electrical and Computer Engineering, NPS-EC-99-002, *PEEB Feedback Control Law Library, Vol. 1: Three-Phase Inverter Control Algorithms*, J.G. Ciezki and R.W. Ashton, pp. 1-42, Unclassified, January, 1999.

[11] Moore, J.E., "Frequency-Based Load Sharing in Current-Mode-Controlled Buck Converters", Master's Thesis, Naval Postgraduate School, Monterey, California, March, 1999.

[12] Tuladhar, A., Jin, H., Unger, T., Mauch, K., "Control of Parallel Inverters in Distributed AC Power Systems with Consideration of the Line Impedence Effect," *Conference Proceedings - IEEE Applied Power Electronics Conference and Exposition - APEC Proceedings of the 1998 13thAnnual Applied Power Electronics Conference and Exposition, APEC'98*, v. 1, pp. 321-328, February1998.

[13] Raymond, David J., "A Radically Moderns Approach to Introductory Physics – Volume 1." [http://www.physics.nmt.edu/raymond/ph131/book1/book1.html]. 1998.

[14] *The ARRL Handbook for the Radio Amateur*, 65th ed., pp. C-63, American Radio Relay League, 1988.

[15] No Author, "Toshiba 4100 Plus Series 10, 25, and 50 kVA Three-Phase UPS." [http://www.carebase.com/41003ph.htm]. May 1999.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center.....2
   8725 John J. Kingman Rd., STE 0944
   Fort Belvoir, Virginia 22060-6218

2. Dudley Knox Library.....2
   Naval Postgraduate School
   411 Dyer Rd.
   Monterey, California 93943-5101

3. Chairman, Code EC.....1
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California 93943-5121

4. Curriculum Officer, Code 34.....1
   Engineering and Technology
   Naval Postgraduate School
   Monterey, California 93943-5109

5. John Ciezki, Code EC/Cy.....3
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California 93943-5121

6. Robert Ashton, Code EC/Ah.....3
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California 93943-5121

7. Naval Surface Warfare Center.....1
   Attn: Roger Cooley
   Carderock Division, Annapolis Detachment
   3A Leggett Circle
   Annapolis, Maryland 21402-5067

8. Mark Marinac..... 2
   106E. 400N.
   Price, Utah 84501